

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

ИСПОЛЬЗОВАНИЕ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ АНАЛИЗА БИОМЕТРИЧЕСКИХ ДАННЫХ НОСИМЫХ УСТРОЙСТВ С ЦЕЛЬЮ ФОРМИРОВАНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ

USING MATHEMATICAL MODELS FOR ANALYZING BIOMETRIC DATA OF WEARABLE DEVICES TO CREATE AN INFORMATION SYSTEM

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

Ignatova Maria Sergeevna
Candidate of Economic Sciences associate professor
RSCI SPIN-code: 7874-8981
email: Riban1@mail.ru
Kuban State Agricultural university, 350044, Russia, Krasnodar, Kalinina, 13

Оксанич Елена Анатольевна
Кандидат экономических наук, доцент
РИНЦ SPIN-код: 3138-8291
email: 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
Kuban State Agricultural university, 350044, Russia, Krasnodar, Kalinina, 13

Авакимян Наталья Николаевна
К.ф.-м.н., доцент
РИНЦ SPIN-код: 6082-4770
email: avnatali@mail.ru
ФГБОУ «Кубанский государственный аграрный университет», 350044, Россия, г. Краснодар, ул. Калинина 13

Avakimyan Natalia Nikolaevna
Cand.Phys.-Math.Sci., associate professor,
RSCI SPIN-code: 6082-4770
email: avnatali@mail.ru
Kuban State Agricultural university, 350044, Russia, Krasnodar, Kalinina, 13

Карлаш Евгений Сергеевич
студент группы ИТ2442
email: jekakarlash29@mail.ru
ФГБОУ «Кубанский государственный аграрный университет», 350044, Россия, г. Краснодар, ул. Калинина 13

Karlash Evgeniy Sergeevich
student of group IT2442
email: jekakarlash29@mail.ru
Kuban State Agricultural university, 350044, Russia, Krasnodar, Kalinina, 13

Носимые устройства – умные часы и фитнес-трекеры – непрерывно собирают биометрические данные пользователей: частоту сердечных сокращений, количество шагов, качество сна, уровень стресса и расход калорий. Однако, они привязаны к конкретному производителю, требуют подключения к интернету и не предоставляют инструментов глубокого анализа с применением искусственного интеллекта.

In today's world, wearable devices—smartwatches and fitness trackers—continuously collect user biometric data, including heart rate, step count, sleep quality, stress levels, and calorie burn. However, existing solutions are tied to a specific manufacturer, require an internet connection, and lack in-depth analysis tools using artificial intelligence. This study compares five existing health monitoring systems (Apple Health, Google Fit, Garmin Connect, Fitbit, and Samsung Health) and the exported data formats of six wearable device manufacturers, identifying their key limitations. Statistical analysis methods for biometric time series were explored and applied, including descriptive statistics, moving average, exponential smoothing, three-level anomaly detection, and Pearson correlation analysis. This article describes the development of a cross-platform desktop information

В работе проведён сравнительный анализ пяти существующих систем мониторинга здоровья (Apple Health, Google Fit, Garmin Connect, Fitbit, Samsung Health) и форматов экспортируемых данных шести производителей носимых устройств, выявлены их ключевые ограничения. Исследованы и применены методы статистического анализа биометрических временных рядов, включая

описательную статистику, скользящее среднее, экспоненциальное сглаживание, трёхуровневое обнаружение аномалий и корреляционный анализ по Пирсону. В статье описана разработка кроссплатформенной десктопной информационной системы HealthWear Analytics на языке Python с использованием фреймворка PySide6, обеспечивающая импорт данных в форматах CSV и JSON от шести типов носимых устройств, интерактивную визуализацию десяти графиков с персонализированными нормами, ИИ-анализ через API провайдеров Groq, OpenAI и DeepSeek и автоматическую генерацию PDF-отчётов. Практическая значимость – разработанная система позволяет пользователям централизованно хранить и анализировать биометрические данные с носимых устройств шести производителей, получать персонализированные ИИ-рекомендации и формировать структурированные медицинские отчёты для передачи лечащему врачу. Система функционирует полностью в офлайн-режиме, обеспечивая конфиденциальность персональных данных

Ключевые слова: МАТЕМАТИЧЕСКИЕ МОДЕЛИ АНАЛИЗА, ИНФОРМАЦИОННАЯ СИСТЕМА, БИОМЕТРИЧЕСКИЕ ДАННЫЕ, БАЗЫ ДАННЫХ, PYTHON

system, HealthWear Analytics, written in Python and using the PySide6 framework. It supports data import in CSV and JSON formats from six types of wearable devices, interactive visualization of ten graphs with personalized norms, AI analysis via API providers Groq, OpenAI, and DeepSeek, and automatic PDF report generation. Practical significance: the developed system allows users to centrally store and analyze biometric data from wearable devices from six manufacturers, receive personalized AI recommendations, and generate structured medical reports for sharing with their physician. The system operates completely offline, ensuring the confidentiality of personal data

Keywords: MATHEMATICAL MODELS OF ANALYSIS, INFORMATION SYSTEM, BIOMETRIC DATA, DATABASES, PYTHON

<http://dx.doi.org/10.21515/1990-4665-220-048>

Введение

Носимые устройства – умные часы и фитнес-трекеры – непрерывно собирают биометрические данные пользователей: частоту сердечных сокращений, количество шагов, качество сна, уровень стресса и расход калорий. Однако, существующие решения привязаны к конкретному производителю, требуют подключения к интернету и не предоставляют инструментов глубокого анализа с применением искусственного интеллекта.

Актуальность работы обусловлена потребностью в едином кроссплатформенном инструменте, способном агрегировать данные от устройств различных производителей и формировать персонализированные рекомендации по улучшению показателей здоровья.

Цель статьи - исследование методов анализа биометрических данных носимых устройств и разработка кроссплатформенной информационной

системы мониторинга показателей здоровья с поддержкой ИИ и автоматической генерацией отчётов.

Для достижения цели в работе решались следующие задачи:

- провести сравнительный анализ существующих систем мониторинга здоровья и форматов экспортируемых данных;
- исследовать и применить статистические методы анализа биометрических временных рядов;
- разработать архитектуру и реализовать информационную систему HealthWear Analytics;
- оценить экономическую эффективность системы.

Обзор и сравнительный анализ существующих систем

На сегодняшний день на рынке представлен широкий спектр информационных систем для работы с данными носимых устройств. Рассмотрим наиболее распространённые из них.

Apple Health – встроенное приложение операционной системы iOS, являющееся центральным хранилищем биометрических данных в экосистеме Apple. Система поддерживает сбор более 80 видов показателей здоровья, интеграцию с Apple Watch и сторонними приложениями через HealthKit API. К достоинствам системы относятся глубокая интеграция с устройствами Apple, богатый набор отслеживаемых показателей и возможность передачи данных лечащему врачу. Основные недостатки: полная привязка к экосистеме Apple, отсутствие поддержки устройств сторонних производителей, ограниченные инструменты аналитики и невозможность использования на платформах Android и Windows.

Google Fit – кроссплатформенная система мониторинга здоровья от компании Google, доступная на устройствах Android и iOS. Система агрегирует данные с совместимых носимых устройств и фитнес-приложений через REST API. Достоинства: кроссплатформенность, открытый API для интеграции, поддержка широкого круга устройств. Недостатки: облачная архитектура с обязательной передачей данных на

серверы Google, ограниченные аналитические возможности, отсутствие ИИ-анализа и поддержки десктопных платформ.

Garmin Connect – фирменная платформа компании Garmin для работы с данными устройств собственного производства. Система предоставляет детальную аналитику спортивных показателей, включая анализ тренировок, состояния восстановления и физической формы. Достоинства: глубокая интеграция с устройствами Garmin, развитые инструменты спортивной аналитики, веб-интерфейс и мобильное приложение. Недостатки: поддержка исключительно устройств Garmin, облачная архитектура, отсутствие ИИ-рекомендаций. Fitbit (Google) – платформа мониторинга здоровья, интегрированная в экосистему Google после приобретения компании Fitbit в 2021 году. Система предоставляет подробную аналитику сна, активности и здоровья сердца. Достоинства: удобный интерфейс, детальный анализ сна, интеграция с Google сервисами. Недостатки: поддержка только устройств Fitbit, платная подписка Fitbit Premium для расширенной аналитики, облачное хранение данных. Samsung Health – платформа мониторинга здоровья от компании Samsung, ориентированная на владельцев устройств Samsung Galaxy. Система поддерживает широкий спектр биометрических показателей и интеграцию с Galaxy Watch. Достоинства: глубокая интеграция с устройствами Samsung, богатый функционал для отслеживания здоровья. Недостатки: ограниченная поддержка устройств сторонних производителей, мобильная платформа без десктопного клиента. Сравнительный анализ рассмотренных систем представлен в таблице 1. Ни одна из рассмотренных систем не обеспечивает одновременно кроссплатформенность, поддержку устройств различных производителей, офлайн-режим работы и ИИ-анализ показателей здоровья.

Биометрические данные носимых устройств, по своей природе, представляют собой временные ряды – последовательности значений показателей здоровья, упорядоченных по времени измерения.

Таблица 1 – Сравнительный анализ существующих систем мониторинга здоровья

Критерий	Apple Health	Google Fit	Garmin Connect	Fitbit	Samsung Health	HealthWear Analytics
Кроссплатформенность	iOS	Android/iOS	Web/Mobile	Web/Mobile	Android	Windows/macOS/Linux
Поддержка устройств разных производителей	Нет	Частично	Нет	Нет	Нет	Да (6 типов)
Офлайн-режим	Частично	Нет	Нет	Нет	Частично	Да
ИИ-анализ	Нет	Нет	Нет	Платно	Нет	Да
Генерация PDF-отчётов	Нет	Нет	Да	Да	Нет	Да
Конфиденциальность данных	Облако Apple	Облако Google	Облако Garmin	Облако Google	Облако Samsung	Локально
Открытый исходный код	Нет	Нет	Нет	Нет	Нет	Да

Применяемые методы исследования

В ходе исследования применяются следующие методы:

1) Методы описательной статистики составляют основу анализа биометрических временных рядов.

2) Метод оценки тренда позволяет определить систематическую составляющую изменения показателя во времени. В системе HealthWear Analytics применяется метод сравнения средних значений первой и второй половин временного ряда. Данный метод прост в реализации и интерпретации, что делает его предпочтительным для включения результатов в контекст ИИ-анализа в виде понятных текстовых описаний, например: «наблюдается рост среднего пульса на 8 уд/мин во второй половине периода».

3) Метод скользящего среднего применяется для сглаживания временного ряда и выявления общей тенденции изменения показателя, свободной от случайных флуктуаций. При анализе суточных данных

носимых устройств рекомендуется использовать окно $k = 7$ (семидневное скользящее среднее), что позволяет сгладить влияние отдельных дней с аномально высокой или низкой активностью и выявить недельный тренд показателя.

4) Анализ суточных паттернов является специфическим методом для биометрических данных с высокой частотой дискретизации. Суточный паттерн представляет собой усреднённый профиль изменения показателя в течение суток, построенный по данным за несколько дней. Для показателя ЧСС типичный суточный паттерн характеризуется минимальными значениями в период с 2:00 до 5:00 (фаза глубокого сна) и максимальными значениями в период дневной активности с 10:00 до 18:00. Существенное отклонение индивидуального суточного паттерна пользователя от типичного является диагностически значимым признаком. В системе HealthWear Analytics демонстрационные данные генерируются с учётом суточного паттерна ЧСС посредством модуляции базового значения синусоидальной функцией с периодом 24 часа.

Метод экспоненциального сглаживания является альтернативой скользящему среднему и обладает важным преимуществом: он придаёт больший вес последним наблюдениям, что делает его более чувствительным к недавним изменениям в состоянии здоровья пользователя. Сравнение метода скользящего среднего и экспоненциального сглаживания применительно к биометрическим данным представлено в таблице 2.

Таблица 2 – Сравнение методов сглаживания временных рядов

Критерий	Скользящее среднее	Экспоненциальное сглаживание
Вес последних наблюдений	Равномерный	Экспоненциально убывающий
Чувствительность к изменениям	Средняя	Высокая
Устойчивость к выбросам	Средняя	Средняя
Требуемый объём данных	к последних значений	Все предыдущие значения
Интерпретируемость	Высокая	Средняя
Применение в системе	Недельный тренд	Оперативный мониторинг

Скользящее среднее применяется для оценки недельного тренда биометрических показателей, а метод экспоненциального сглаживания используется при формировании контекста ИИ-анализа для выявления краткосрочных изменений в состоянии здоровья.

Обнаружение аномалий является одной из ключевых задач анализа биометрических данных. Аномалия определяется как значение показателя, существенно отклоняющееся от нормального диапазона или типичного поведения временного ряда. С медицинской точки зрения аномалии биометрических показателей могут свидетельствовать о нарушениях здоровья, требующих внимания специалиста.

В системе HealthWear Analytics применяется трёхуровневый подход к обнаружению аномалий, сочетающий методы на основе знаний о предметной области и статистические методы.

Первый уровень – метод персонализированных пороговых значений. Данный метод основан на сравнении измеренного значения показателя с физиологически обоснованными границами нормы, персонализированными по возрасту и полу пользователя.

Персонализированные пороги для показателя ЧСС представлены в таблице 3.

Таблица 3 – Персонализированные пороговые значения ЧСС

Возрастная группа	Нижняя граница (уд/мин)	Верхняя граница (уд/мин)	Критическое значение (уд/мин)
До 18 лет	60	100	110
18-40 лет	55	90	105
40-60 лет	55	95	110
Старше 60 лет	60	100	115

Для женщин верхняя граница нормы и критическое значение увеличиваются на 5 уд/мин. Аналогичным образом персонализированы пороги для остальных показателей. Для продолжительности сна нормальный диапазон составляет 420–540 минут (7–9 часов) для взрослых 18–65 лет, 480–600 минут для лиц моложе 18 лет и 390–480 минут для лиц старше 65 лет. Для двигательной активности норма составляет не менее 8

000 шагов в сутки для взрослых до 65 лет и не менее 6 000 шагов для лиц старше 65 лет. Для уровня стресса предупредительный порог установлен на уровне 60 баллов, критический – на уровне 75 баллов. Для SpO2 норма составляет не менее 96%, предупредительный порог – 94%. Второй уровень – метод статистических границ, основанный на правиле трёх сигм. Значение считается статистически аномальным, если оно отклоняется от среднего значения ряда более чем на три среднеквадратических отклонения. Согласно нормальному распределению, за пределами трёх сигм находится не более 0,3% значений. Третий уровень – метод интегральных флагов отклонений. Данный метод представляет собой интегральную оценку состояния здоровья пользователя на основе совокупности показателей за анализируемый период. В системе определены пять бинарных флагов: повышенный пульс (high_hr), высокий стресс (high_stress), недостаток сна (low_sleep), низкий SpO2 (low_spo2), малая двигательная активность (low_steps). Флаг устанавливается в значение True, если среднее значение соответствующего показателя за анализируемый период выходит за пределы нормального диапазона, определённого на первом уровне.

Теоретической основой корреляционного анализа биометрических данных служат известные физиологические взаимосвязи. Между ЧСС и уровнем стресса существует положительная корреляция: активация симпатической нервной системы при стрессе вызывает учащение сердечного ритма. Между продолжительностью сна и уровнем стресса существует отрицательная корреляция: хронический недосып повышает уровень кортизола и, как следствие, уровень стресса. Между двигательной активностью и ЧСС в состоянии покоя существует отрицательная долгосрочная корреляция: регулярная физическая активность снижает частоту сердечного ритма в покое благодаря адаптации сердечно-сосудистой системы.

Для оценки силы и направления линейной связи между двумя временными рядами применяется коэффициент корреляции Пирсона.

Коэффициент корреляции Пирсона принимает значения в диапазоне $[-1, 1]$. Интерпретация значений коэффициента применительно к биометрическим данным представлена в таблице 4.

Таблица 4 – Интерпретация коэффициента корреляции Пирсона

Значение $ r $	Интерпретация	Пример для биометрических данных
Менее 0,25	Связь отсутствует	Показатели независимы друг от друга
От 0,25 до 0,50	Слабая корреляция	Незначительная взаимосвязь
От 0,50 до 0,75	Умеренная корреляция	Умеренная взаимосвязь ЧСС и стресса
От 0,75 до 1	Сильная корреляция	Тесная взаимосвязь сна и стресса

Знак коэффициента определяет направление связи. Положительная корреляция ($r > 0$) между ЧСС и уровнем стресса означает, что повышение стресса сопровождается учащением пульса. Отрицательная корреляция ($r < 0$) между продолжительностью сна и уровнем стресса означает, что недостаток сна связан с повышением уровня стресса.

В системе HealthWear Analytics корреляционный анализ выполняется при наличии не менее четырёх измерений в каждом временном ряду и только для пар рядов одинаковой длины. Пороговое значение для признания корреляции значимой установлено на уровне $|r| \geq 0,25$. Вычисление коэффициентов корреляции реализовано с использованием функции `numpy.corrcoef`. Результаты корреляционного анализа формируются в виде структурированных текстовых описаний с указанием значения коэффициента и интерпретации направления связи, например: «Пульс и стресс: корреляция 0,54 (прямая связь)», «Сон и стресс: корреляция $-0,61$ (обратная связь)». Данные описания включаются в контекст запроса к ИИ-модели, что позволяет формировать рекомендации с учётом индивидуальных взаимосвязей между показателями здоровья конкретного пользователя.

Выбор математических моделей для системы HealthWear Analytics определялся рядом критериев: соответствием физиологической природе анализируемых данных, вычислительной эффективностью для обработки данных в реальном времени, интерпретируемостью результатов для формирования понятных пользователю рекомендаций, а также

возможностью интеграции результатов в контекст ИИ-анализа в текстовом формате. В таблице 5 представлен сравнительный анализ рассмотренных методов анализа биометрических данных.

Таблица 5 – Сравнительный анализ методов анализа биометрических данных

Метод	Вычислительная сложность	Интерпретируемость	Устойчивость к выбросам	Персонализация
Описательная статистика	$O(n)$	Высокая	Средняя	Нет
Медиана	$O(n \log n)$	Высокая	Высокая	Нет
Скользящее среднее	$O(n)$	Высокая	Средняя	Нет
Пороговые значения	$O(n)$	Высокая	Высокая	Да
Правило трёх сигм	$O(n)$	Средняя	Низкая	Да
Корреляция Пирсона	$O(n)$	Высокая	Низкая	Нет

Все рассмотренные методы имеют линейную или близкую к линейной вычислительной сложности $O(n)$, что обеспечивает их эффективное применение для обработки временных рядов объёмом до нескольких тысяч измерений без заметных задержек. Высокая интерпретируемость результатов является ключевым требованием для системы мониторинга здоровья, ориентированной на широкую аудиторию, и обеспечивается всеми выбранными методами.

Более сложные методы анализа временных рядов, такие как модели ARIMA, алгоритмы кластеризации или нейронные сети, не были включены в систему по ряду причин. Во-первых, они требуют значительно большего объёма обучающих данных, чем может предоставить один пользователь носимого устройства. Во-вторых, их результаты сложнее интерпретировать и включить в текстовый контекст ИИ-анализа. В-третьих, роль интеллектуального анализа в системе выполняют генеративные ИИ-модели (LLaMA 3.3, GPT-4o-mini, DeepSeek), которым передаются структурированные статистические характеристики, что делает применение отдельных моделей машинного обучения избыточным.

Таким образом, выбранная совокупность методов – описательная статистика, оценка тренда, скользящее среднее, трёхуровневое обнаружение аномалий и корреляционный анализ Пирсона – обеспечивает комплексный, вычислительно эффективный и интерпретируемый анализ биометрических данных носимых устройств, соответствующий требованиям системы HealthWear Analytics.

Разработка информационной системы Healthwear Analytics

Архитектура информационной системы HealthWear Analytics построена на принципе многоуровневого разделения ответственности (Separation of Concerns), при котором каждый модуль системы отвечает за строго определённый круг задач и не содержит логики, относящейся к смежным уровням. Данный принцип обеспечивает низкую связность компонентов, упрощает тестирование и сопровождение системы.

Архитектура системы включает четыре уровня. Общая структурная схема архитектуры представлена на рисунке 1. На диаграмме отображены все четыре уровня системы с перечнем входящих в каждый уровень модулей и направления зависимостей между уровнями.

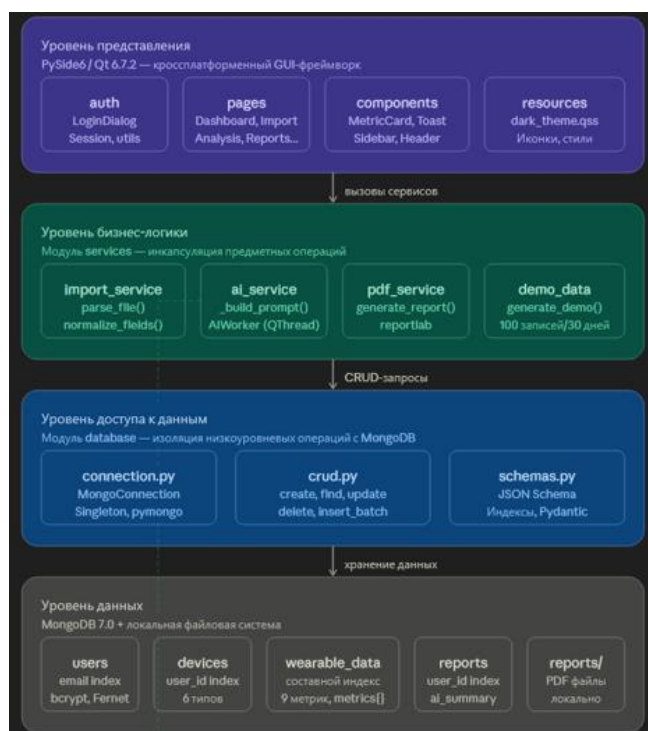


Рисунок 1 – Архитектура информационной системы HealthWear Analytics

Уровень представления реализован с использованием фреймворка PySide6 – официальной Python-обёртки над библиотекой Qt 6. Данный уровень включает страницы пользовательского интерфейса (модуль `pages`), переиспользуемые UI-компоненты (модуль `components`) и диалоговые окна аутентификации (модуль `auth`). Уровень представления не содержит логики работы с базой данных и обращается к нижележащим уровням исключительно через определённые интерфейсы.

Уровень бизнес-логики реализован в модуле `services` и включает четыре сервиса: сервис импорта данных (`import_service`), сервис генерации демонстрационных данных (`demo_data_service`), сервис ИИ-анализа (`ai_service`) и сервис генерации PDF-отчётов (`pdf_service`). Каждый сервис инкапсулирует логику соответствующей предметной операции и не зависит от конкретной реализации уровня представления.

Уровень доступа к данным реализован в модуле `database` и включает три компонента: модуль подключения к MongoDB (`connection`), модуль CRUD-операций (`crud`) и модуль схем и валидаторов (`schemas`). Весь низкоуровневый доступ к базе данных инкапсулирован в модуле `crud`, что позволяет при необходимости заменить СУБД без изменения вышележащих уровней.

Уровень данных представлен базой данных MongoDB с четырьмя коллекциями, описанными в главе 3, а также локальной файловой системой для хранения сгенерированных PDF-отчётов.

Структура каталогов проекта HealthWear Analytics представлена на рисунке 2. Корневой каталог содержит главный модуль запуска `main.py`, файл конфигурации `config.py` и модуль настройки логирования `logging_config.py`. Подкаталоги `auth`, `components`, `database`, `models`, `pages`, `services` и `resources` содержат соответствующие модули системы. Каждый подкаталог содержит файл `__init__.py`, что делает его полноценным Python-пакетом.

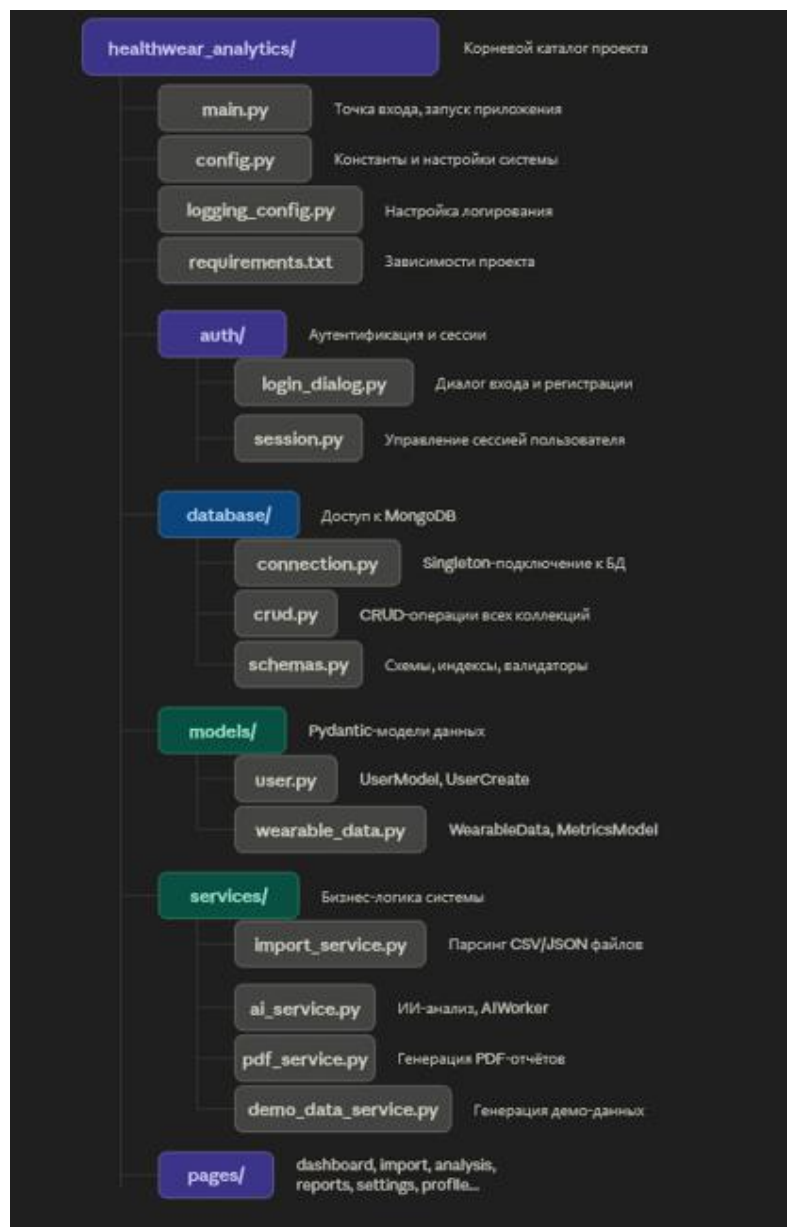


Рисунок 2 – Структура каталогов проекта HealthWear Analytics

Для реализации системы выбран технологический стек, представленный в таблице 6.

Таблица 6 – Технологический стек системы HealthWear Analytics

Компонент	Технология	Версия	Назначение
Язык программирования	Python	3.11+	Основной язык разработки
GUI-фреймворк	PySide6	6.7.2	Кроссплатформенный интерфейс
База данных	MongoDB	7.0+	Хранение биометрических данных
Драйвер БД	Pymongo	4.8.0	Взаимодействие с MongoDB
Валидация данных	Pydantic	2.8.2	Валидация на уровне приложения
Визуализация	matplotlib	3.9.2	Графики дашборда и аналитики
Интерактивные графики	Plotly	5.23.0	Расширенная аналитика
Генерация PDF	reportlab	4.2.2	Формирование отчётов
ИИ-провайдер 1	Groq (LLaMA 3.3)	0.11.0	ИИ-анализ по умолчанию
ИИ-провайдер 2	OpenAI (GPT-4o-mini)	1.51.2	Альтернативный ИИ-анализ
Хеширование паролей	bcrypt	4.2.0	Безопасное хранение паролей
Шифрование	cryptography (Fernet)	43.0.1	Шифрование API-ключей
Обработка данных	pandas	2.2.2	Работа с табличными данными
Математические вычисления	numpy	–	Статистический анализ
Тестирование	pytest	8.3.2	Модульное тестирование

Выбор PySide6 в качестве GUI-фреймворка обусловлен его кроссплатформенностью – приложение без изменений кода запускается на Windows, macOS и Linux, – а также богатым набором встроенных виджетов, поддержкой стилизации через таблицы стилей QSS и возможностью встраивания matplotlib-графиков непосредственно в интерфейс через FigureCanvasQTAagg.

Модуль импорта и хранения данных обеспечивает загрузку биометрических данных из файлов CSV и JSON в базу данных системы. Модуль реализован в двух компонентах: сервисе парсинга файлов

(services/import_service.py) и странице пользовательского интерфейса импорта (pages/import_page.py).

Алгоритм импорта данных представлен на рисунке 3. На блок-схеме отображены последовательные шаги обработки файла начиная с момента его выбора пользователем: определение формата файла, парсинг, нормализация полей, валидация через Pydantic, пакетная вставка в MongoDB и уведомление пользователя о результате. Также отображены ветки обработки ошибок на каждом этапе. Сервис парсинга файлов реализует функцию `parse_file`, которая принимает путь к файлу и тип устройства и возвращает список словарей с биометрическими записями в унифицированном формате. Унификация форматов различных производителей является ключевой задачей сервиса, поскольку каждый производитель использует собственные наименования полей и единицы измерения. Например, Apple Watch экспортирует данные о пульсе в поле `heartRate`, Fitbit – в поле `heart_rate`, Garmin – в поле `avg_hr`. Сервис нормализует все эти варианты к единому полю `heart_rate` внутренней схемы данных.

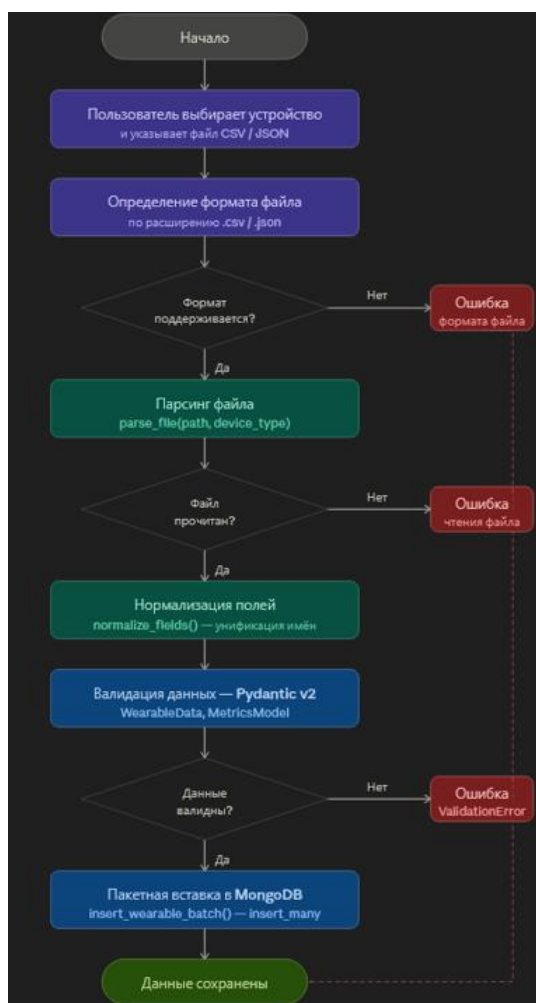


Рисунок 3 – Блок-схема алгоритма импорта данных

После успешного парсинга файла данные передаются в функцию `insert_wearable_batch` модуля `database/crud.py`, которая выполняет пакетную вставку записей в коллекцию `wearable_data`. Пакетная вставка реализована через метод `insert_many` драйвера `pyMongo`, что существенно повышает производительность по сравнению с последовательной вставкой отдельных документов.

Помимо импорта из файлов, модуль поддерживает генерацию демонстрационных данных. Сервис `demo_data_service` формирует ровно 100 биометрических записей за последние 30 дней с реалистичными значениями показателей. Для обеспечения реалистичности применяется суточный паттерн ЧСС: базовое значение пульса модулируется синусоидальной функцией с периодом 24 часа.

Страница импорта данных в интерфейсе приложения представлена на рисунке 4. Страница предоставляет пользователю возможности: выбор зарегистрированного устройства, загрузка файла CSV или JSON, генерация демонстрационных данных, просмотр загруженных записей в табличном виде с пагинацией, поиск по таблице и удаление записей.

Дата и время	Устройство	Шаги	Ср. пульс	SpO2	Сон	Глубокий сон	Стресс	Действия
2026-04-22 20:31	Apple Watch	8318	62	96	271	29	75	Удалить
2026-04-21 22:07	Apple Watch	12290	75	97	271	26	69	Удалить
2026-04-20 23:40	Apple Watch	9062	76	96	368	52	47	Удалить
2026-04-19 22:21	Apple Watch	9639	73	96	516	96	25	Удалить
2026-04-18 20:30	Apple Watch	10848	72	97	473	80	19	Удалить
2026-04-17 23:03	Apple Watch	10889	73	96	323	40	66	Удалить
2026-04-16 21:53	Apple Watch	18702	89	96	402	52	43	Удалить
2026-04-15 20:35	Apple Watch	11273	77	97	324	54	33	Удалить
2026-04-14 23:09	Apple Watch	10300	68	97	467	64	43	Удалить
2026-04-13 23:06	Apple Watch	1487	80	96	503	42	66	Удалить
2026-04-12 20:26	Apple Watch	6639	64	97	471	56	20	Удалить
2026-04-11 22:25	Apple Watch	7655	68	97	401	50	21	Удалить
2026-04-10 23:34	Apple Watch	12333	77	97	396	51	47	Удалить
2026-04-09 22:41	Apple Watch	51404	78	96	570	16	67	Удалить

Рисунок 4 – Страница импорта данных в интерфейсе приложения

Таблица записей реализована с использованием класса `PandasTableModel` – адаптера между `pandas DataFrame` и виджетом `QTableView`. Пагинация реализована на уровне запросов к базе данных с использованием методов `skip` и `limit` драйвера `pyMongo`, что обеспечивает загрузку только текущей страницы из 20 записей без извлечения всего набора в память.

Модуль визуализации и дашборда обеспечивает наглядное представление биометрических данных пользователя в виде карточек ключевых метрик, интерактивных графиков и аналитических блоков. Модуль реализован в странице дашборда (`pages/dashboard_page.py`) и странице аналитики (`pages/analysis_page.py`).

Страница дашборда отображает агрегированные показатели здоровья за выбранный период. В верхней части страницы расположены шесть карточек метрик, реализованных в компоненте MetricCard. Каждая карточка отображает название показателя, его среднее значение за период, единицу измерения, минимальное и максимальное значения, а также цветовой индикатор соответствия норме. Карточки отображают следующие показатели: средний пульс (уд/мин), суммарное количество шагов, средний уровень SpO2 (%), среднее время сна (мин), уровень стресса и суммарный расход калорий (ккал). В нижней части дашборда расположены два графика на основе matplotlib: столбчатая диаграмма динамики ежедневного количества шагов и кольцевая диаграмма структуры сна с разбивкой по фазам. Страница аналитики является центральным аналитическим экраном системы и предоставляет детальное многоуровневое представление биометрических данных пользователя. Внешний вид страницы представлен на рисунках 5 и 6.



Рисунок 5 – Страница аналитики: графики динамики биометрических показателей

В верхней части страницы расположена панель управления с выбором периода анализа (День, Неделя, Месяц) и фильтром по устройству, а также кнопки «Обновить», «ИИ-анализ» и «PDF-отчёт». Ниже панели управления

отображаются пять карточек сводной статистики с указанием среднего, минимального и максимального значений за период для показателей пульса, шагов, SpO2, сна и стресса. Под карточками расположена информационная строка, указывающая возраст и пол пользователя, на основе которых рассчитаны персонализированные нормы, например: «Нормы рассчитаны для возраста 25 лет · Мужской».



Рисунок 6 – Страница аналитики: графики SpO2 и распределения пульса, результаты ИИ-анализа

Блок «Взаимосвязи показателей» отображает результаты корреляционного анализа в форме понятных текстовых описаний. Каждая выявленная закономерность содержит описание связи между двумя показателями, процент совпадений и практическую рекомендацию.

Блок «Влияние стресса на показатели» отражает персонализированные числовые данные о том, как эпизоды высокого стресса влияют на остальные биометрические показатели конкретного пользователя. Результаты представляются в формате конкретных числовых изменений, например: «При высоком стрессе пульс в среднем выше на 7.0 уд/мин (79 → 86 уд/мин)», «При высоком стрессе сон короче на 114 мин (419 → 305 мин)», «При высоком стрессе активность выше на 1385 шагов

(14001 → 15386 шагов)». Данный блок является наглядным подтверждением результатов корреляционного анализа и формирует доказательную базу для ИИ-рекомендаций.

Блок уведомлений об аномалиях реализован в виде кнопки-бейджа в заголовке страницы, отображающей количество выявленных аномалий за анализируемый период. При нажатии на бейдж открывается модальное диалоговое окно с детальным перечнем аномальных измерений, сгруппированных по показателям, с указанием даты, значения и величины отклонения от персонализированной нормы.

Ниже аналитических блоков расположены 10 графиков, выполненных с использованием библиотеки `matplotlib` и расположенных попарно в пять рядов, в частности:

- первый график – «Динамика пульса» – представляет собой линейный график ЧСС с двумя горизонтальными пунктирными линиями верхней и нижней границ нормального диапазона;

- второй график – «Шаги по дням» – площадный график ежедневной двигательной активности с линией рекомендуемого минимума в 8 000 шагов;

- третий график – «Структура сна» – площадный составной график с разбивкой по трём фазам сна: глубокий, лёгкий и REM-сон, что позволяет оценить не только общую продолжительность сна, но и его качество.

Все графики оформлены в единой тёмной цветовой схеме: фон – `#1e2028`, основной цвет текста – `#c9ccd6`, акцентный цвет – `#5b8dee`. Границы нормального диапазона на всех графиках являются персонализированными и вычисляются функцией `_get_norms` на основе возраста и пола пользователя. При изменении фильтров периода и устройств главное окно `MainWindow` инициирует повторную выборку данных и перерисовку всех элементов страницы через метод `on_filters_changed`.

Тестирование информационной системы

Тестирование информационной системы HealthWear Analytics проводилось на двух уровнях: модульное тестирование с использованием фреймворка pytest и функциональное тестирование в режиме опытной эксплуатации.

Модульные тесты реализованы в файле tests/test_smoke.py и охватывают три группы проверок. Первая группа – тест импорта всех модулей системы – проверяет, что все 25 модулей приложения успешно импортируются без ошибок. Данный тест выявляет синтаксические ошибки, проблемы с зависимостями и циклические импорты на этапе сборки. Вторая группа – тест подключения к MongoDB с использованием Mock-объектов – проверяет логику класса MongoConnection без реального подключения к базе данных через механизм monkeypatch фреймворка pytest. Третья группа – тест создания пользователя и генерации демонстрационных данных – проверяет бизнес-логику функций create_user и generate_demo_records.

Результаты функционального тестирования основных сценариев использования представлены в таблице 7.

Таблица 7 – Результаты функционального тестирования

Сценарий тестирования	Ожидаемый результат	Фактический результат	Статус
Регистрация нового пользователя	Создание учётной записи, вход в систему	Соответствует ожидаемому	Пройден
Вход с неверным паролем	Сообщение об ошибке аутентификации	Соответствует ожидаемому	Пройден
Импорт CSV-файла Apple Watch	Загрузка записей в БД, отображение в таблице	Соответствует ожидаемому	Пройден
Импорт JSON-файла Fitbit	Загрузка записей в БД, отображение в таблице	Соответствует ожидаемому	Пройден
Импорт файла неверного формата	Сообщение об ошибке без завершения работы	Соответствует ожидаемому	Пройден
Генерация демонстрационных данных	Создание 100 записей за 30 дней	Соответствует ожидаемому	Пройден
Отображение дашборда	Карточки метрик и графики за выбранный период	Соответствует ожидаемому	Пройден
Фильтрация по периоду и устройству	Обновление графиков и метрик	Соответствует ожидаемому	Пройден
ИИ-анализ через Groq	Получение JSON-заключения, отображение в интерфейсе	Соответствует ожидаемому	Пройден
ИИ-анализ без API-ключа	Информативное сообщение об ошибке	Соответствует ожидаемому	Пройден
Генерация PDF-отчёта	Создание файла, сохранение в БД	Соответствует ожидаемому	Пройден
Смена пароля	Обновление хеша в БД, повторная аутентификация	Соответствует ожидаемому	Пройден
Удаление устройства	Удаление устройства и всех его записей	Соответствует ожидаемому	Пройден
Отображение при отсутствии данных	Информативное сообщение вместо пустых графиков	Соответствует ожидаемому	Пройден

По результатам тестирования все 14 проверенных сценариев использования выполнены успешно. Система корректно обрабатывает

граничные случаи и отображает информативные сообщения об ошибках без аварийного завершения работы. Модульные тесты реализованы в файле `tests/test_smoke.py` и охватывают три группы проверок. Первая группа – тест импорта всех модулей системы – проверяет, что все 25 модулей приложения успешно импортируются без ошибок. Данный тест выявляет синтаксические ошибки, проблемы с зависимостями и циклические импорты на этапе сборки. Вторая группа – тест подключения к MongoDB с использованием Mock-объектов – проверяет логику класса `MongoConnection` без реального подключения к базе данных через механизм `monkeypatch` фреймворка `pytest`. Третья группа – тест создания пользователя и генерации демонстрационных данных – проверяет бизнес-логику функций `create_user` и `generate_demo_records`.

Оценка практической и экономической эффективности системы

Одним из важных критериев качества разрабатываемой информационной системы является её практическая и экономическая эффективность. Проведение оценки эффективности позволяет определить степень достижения поставленных целей автоматизации, установить практическую значимость разработанного программного продукта и обосновать целесообразность его использования.

До использования разработанной системы пользователю необходимо самостоятельно выполнять анализ экспортированных данных, полученных из программного обеспечения производителей носимых устройств. При этом требуется проводить изучение показателей за различные периоды времени, выполнять сравнение результатов наблюдений, выявлять возможные отклонения и подготавливать итоговые отчёты. Такой подход требует значительных временных затрат и увеличивает вероятность возникновения ошибок, связанных с человеческим фактором.

Для оценки экономической эффективности разработанной системы необходимо определить затраты на её разработку и ожидаемый экономический эффект от внедрения. Разработка программного продукта

выполнялась одним разработчиком в течение трёх месяцев. Среднемесячная заработная плата разработчика была принята равной 80 000 рублей.

Рассчитаем фонд оплаты труда: $\Phi OT = 80000 \times 3 = 240\,000$ руб.

Страховые отчисления: $C_{отч} = 240000 \times 0,30 = 72000$ руб

В процессе разработки использовался персональный компьютер мощностью 0,3 кВт. Средняя продолжительность работы составляла 4 часа в день. Общая продолжительность разработки составила 90 дней. Общее время работы оборудования: $T = 90 \times 4 = 360$ ч

Затраты на электроэнергию: $\mathcal{E} = 0,3 \times 360 \times 6 = 648$ руб.

Для оценки амортизационных расходов примем стоимость компьютера равной 90 000 рублей, а срок полезного использования оборудования – 5 лет.

Годовая амортизация: $A_r = \frac{90\,000}{5} = 18\,000$ руб.

Поскольку разработка выполнялась в течение трёх месяцев, амортизационные расходы составят: $A = \frac{A_r}{12 \times 3} = 4\,500$ руб.

Результаты расчета затрат на разработку представлены в таблице 8.

Таблица 8 – Затраты на разработку системы.

Статья затрат	Сумма, руб.
Оплата труда разработчика	240 000
Страховые отчисления	72 000
Электричество	648
Амортизация оборудования	4 500
Итого	317 148

После определения затрат на разработку необходимо оценить эффективность использования системы.

До внедрения системы пользователю необходимо самостоятельно анализировать биометрические данные, выявлять возможные отклонения показателей, выполнять интерпретацию результатов наблюдений и

подготавливать итоговые отчёты. После внедрения системы большая часть перечисленных операций выполняется автоматически.

Сравнение временных затрат представлено в таблице 9.

Таблица 9 – Сравнение временных затрат при обработке данных

Операция	До внедрения, мин.	После внедрения, мин.
Анализ биометрических показателей	30	5
Выявление аномалий и отклонений	15	1
Формирование отчета	20	2
Подготовка рекомендаций	25	2
Итого	90	10

Экономия времени определяется как: $T_э = 90 - 10 = 80$ мин.

Относительное сокращение трудозатрат: $K_э = \frac{80}{90} \times 100\% = 88,9\%$

Полученный результат показывает, что использование разработанной системы позволяет сократить временные затраты на обработку биометрических данных практически на 89 %.

Для оценки экономического эффекта предположим, что пользователь выполняет в среднем двадцать циклов анализа данных в месяц. Тогда годовая экономия времени составит:

$$T_{\text{год}} = 80 \times 20 \times 12 = 19\,200 \text{ мин.} = 320 \text{ ч.}$$

При условной стоимости одного часа работы пользователя, равной 500 руб., годовой экономический эффект составит: $Э_{\text{год}} = 320 \times 500 = 160\,000 \text{ руб.}$

$$\text{Срок окупаемости системы: } T_{\text{ок}} = \frac{317\,148}{160\,000} = 1,98 \text{ года}$$

Помимо количественных показателей разработанная система обеспечивает ряд качественных преимуществ. Использование системы позволяет автоматизировать процессы анализа биометрических данных, сократить вероятность возникновения ошибок при обработке информации, повысить наглядность представления результатов за счёт визуализации

данных и обеспечить автоматическое формирование аналитических отчётов.

Дополнительным преимуществом является применение алгоритмов искусственного интеллекта для формирования рекомендаций на основе результатов анализа. В отличие от традиционного подхода пользователь получает не только статистическую информацию, но и интерпретацию полученных данных, что повышает практическую ценность программного продукта.

Таким образом, разработанная информационная система HealthWear Analytics обеспечивает существенное сокращение трудозатрат на обработку биометрических данных, позволяет автоматизировать процессы анализа информации и подготовки отчётов, а также обеспечивает годовой экономический эффект в размере 160 000 рублей. Полученные результаты подтверждают практическую значимость разработанной системы и эффективность её применения для анализа данных, полученных от носимых устройств мониторинга здоровья.

Заключение

Таким образом, для статистической обработки временных рядов применены методы описательной статистики, оценки тренда и скользящего среднего. Для обнаружения аномалий разработан трёхуровневый подход, включающий метод персонализированных пороговых значений, метод статистических границ на основе правила трёх сигм и метод интегральных флагов отклонений. Для выявления взаимосвязей между показателями здоровья применён коэффициент корреляции Пирсона.

В ходе исследования спроектированы инфологическая и даталогическая модели базы данных системы. Разработана схема четырёх коллекций MongoDB – users, devices, wearable_data и reports – с валидаторами JSON Schema и оптимизированными индексами. Реализована двухуровневая валидация данных: на уровне приложения средствами

Pydantic v2 и на уровне базы данных посредством JSON Schema валидаторов MongoDB.

Разработана кроссплатформенная десктопная информационная система HealthWear Analytics на языке Python с использованием фреймворка PySide6. Система реализует полный цикл работы с биометрическими данными: импорт данных в форматах CSV и JSON от шести типов носимых устройств, хранение в MongoDB, интерактивную визуализацию десяти графиков с персонализированными нормами, ИИ-анализ через API трёх провайдеров с формированием структурированного заключения и автоматическую генерацию PDF-отчётов.

Практическая значимость разработанной системы состоит в том, что она позволяет пользователям централизованно хранить и анализировать биометрические данные с носимых устройств различных производителей, получать персонализированные ИИ-рекомендации с опорой на конкретные числовые данные и формировать структурированные медицинские отчёты для передачи лечащему врачу. Система функционирует полностью в офлайн-режиме, обеспечивая конфиденциальность персональных медицинских данных. Необходимо выделить следующие ограничения исследования:

- система не собирает данные автоматически, только импортирует файлы;
- точность анализа зависит от точности исходных данных носимого устройства;
- ИИ-рекомендации не являются медицинским диагнозом;
- поддерживаются только шесть типов устройств;
- требуется интернет для ИИ-анализа (хотя сама система офлайн).

Направлениями дальнейшего развития системы являются: реализация облачной синхронизации данных между устройствами пользователя, расширение поддерживаемых форматов импорта, разработка мобильного

клиента, а также интеграция локальных ИИ-моделей для обеспечения полной автономности анализа без подключения к интернету.

Список использованной литературы:

1. Рамальо, Л. Python – к вершинам мастерства: лаконичное и эффективное программирование / Л. Рамальо ; перевод А. А. Слинкин. – 2-е изд. – Москва : ДМК Пресс, 2022. – 898 с.
2. Стивенсон, Б. Python : сборник упражнений / Б. Стивенсон ; перевод А. Ю. Гинько. – Москва : ДМК Пресс, 2021. – 238 с.
3. Уилкс, М. Профессиональная разработка на Python / М. Уилкс ; перевод А. А. Слинкин. – Москва : ДМК Пресс, 2021. – 502 с. – ISBN 978-5-97060-930-9.
4. Хабаров, А. Н. Разработка программных приложений : учебник / А. Н. Хабаров, А. Н. Ермакова. – Ставрополь : АГРУС, 2025.
5. Маркин, А. В. Постреляционные базы данных. MongoDB : учебное пособие / А. В. Маркин. – 3-е изд. – Москва : Ай Пи Ар Медиа, 2024. – 383 с.
6. Волк, В. К. Базы данных : учебник / В. К. Волк, В. Ю. Осеев, О. С. Черепанов. – Москва, Вологда : Инфра-Инженерия, 2025. – 544 с.
7. Брэдшоу, Ш. MongoDB: полное руководство. Мощная и масштабируемая система управления базами данных / Ш. Брэдшоу, Й. Брэзил, К. Ходоров ; перевод Д. А. Беликов. – Москва : ДМК Пресс, 2020. – 540 с.
8. Бэнкер, К. MongoDB в действии / К. Бэнкер ; перевод А. А. Слинкин. – 2-е изд. – Москва : ДМК Пресс, 2023. – 394 с.
9. Зуев, С. В. Методы анализа данных : учебное пособие / С. В. Зуев. – Москва : Ай Пи Ар Медиа, 2023. – 132 с.
10. Наумов, В. Н. Анализ данных и машинное обучение. Методы и инструментальные средства : учебное пособие / В. Н. Наумов. – Санкт-Петербург : Северо-Западный институт управления РАНХиГС, 2020. – 260 с.
11. Запечников, С. В. Основы интеллектуального анализа данных и машинного обучения: конспект лекций : учебное пособие / С. В. Запечников. – Москва : Национальный исследовательский ядерный университет «МИФИ», 2022. – 136 с.
12. Горбаченко, В. И. Машинное обучение: настраиваем ПО, готовим данные, анализируем / В. И. Горбаченко, К. Е. Савенков, М. А. Малахов. – Москва, Алматы : Ай Пи Ар Медиа, EDP Hub (Идипи Хаб), 2024. – 248 с.
13. Белов В.В. Проектирование информационных систем : учебник / Белов В.В., Чистякова В.И.. – Москва : КУРС, 2024. – 400 с.
14. Гринченко Н.Н. Проектирование информационных систем : учебник / Гринченко Н.Н., Громов А.Ю., Хизриева Н.И.. – Москва : КУРС, 2024. – 176 с.
15. Циперман, Г. Н. Проектирование информационных систем : практикум / Г. Н. Циперман, Н. А. Акатова. – Москва : Издательский Дом МИСиС, 2024. – 200 с.
16. Александровская, Ю. П. Информационные технологии статистического анализа данных : учебно-методическое пособие / Ю. П. Александровская. – Казань : Казанский национальный исследовательский технологический университет, 2019. – 152 с.
17. Дямина, Э. И. Статистический анализ данных с помощью программных средств : практикум / Э. И. Дямина, Л. Н. Титова, А. С. Филиппова. – Саратов : Вузовское образование, 2022. – 98 с.
18. Кубеков Б.С. Модели системы в инженерии программного обеспечения : учебное пособие / Кубеков Б.С.. – Алматы, Москва : EDP Hub (Идипи Хаб), Ай Пи Ар Медиа, 2026. – 359 с.
19. Афонин, Д. Н. Технологии больших данных : учебник / Д. Н. Афонин, П. Н. Афонин. – Санкт-Петербург : Интермедия, 2026. – 104 с.
20. Быстренина, И. Е. Проектирование информационных систем : учебное пособие для вузов / И. Е. Быстренина. – Москва : Дашков и К, 2026. – 118 с.

21. Мартынов, А. П. Информационная безопасность и защита информации : учебное пособие / А. П. Мартынов, И. А. Мартынова, А. А. Русаков. – 2-е изд. – Москва : Ай Пи Ар Медиа, 2026. – 130 с.
22. Меликов, П. И. Изучаем основы Python. Практический курс для дата-аналитиков / П. И. Меликов. – Москва, Алматы : Ай Пи Ар Медиа, EDP Hub (Идипи Хаб), 2026. – 480 с.
23. Шустров, А. С. Основы прикладного программирования на языке Python : учебное пособие / А. С. Шустров. – Москва : Ай Пи Ар Медиа, 2026. – 91 с.
24. Apple Inc. Apple Health – платформа мониторинга здоровья [Электронный ресурс]. – Режим доступа: <https://www.apple.com/health/> (дата обращения: 15.03.2025).
25. Google LLC. Google Fit – платформа мониторинга физической активности [Электронный ресурс]. – Режим доступа: <https://www.google.com/fit/> (дата обращения: 15.03.2025).
26. Garmin Ltd. Garmin Connect – платформа анализа спортивных показателей [Электронный ресурс]. – Режим доступа: <https://connect.garmin.com/> (дата обращения: 15.03.2025).
27. MongoDB Inc. MongoDB 7.0 Documentation – документация системы управления базами данных [Электронный ресурс]. – Режим доступа: <https://www.mongodb.com/docs/> (дата обращения: 15.03.2025).
28. The Qt Company. PySide6 Documentation – документация фреймворка разработки графических приложений [Электронный ресурс]. – Режим доступа: <https://doc.qt.io/qtforpython/> (дата обращения: 15.03.2025).
29. Statista GmbH. Wearable Technology – статистика мирового рынка носимых устройств [Электронный ресурс]. – Режим доступа: <https://www.statista.com/topics/1556/wearable-technology/> (дата обращения: 15.03.2025).
30. Groq Inc. Groq API Documentation – документация программного интерфейса для работы с языковыми моделями [Электронный ресурс]. – Режим доступа: <https://console.groq.com/docs/> (дата обращения: 15.03.2025).

References:

1. Ramal'o, L. Python – k veršinam masterstva: lakoničnoe i jeffektivnoe programmirovanie / L. Ramal'o ; perevod A. A. Slinkin. – 2-e izd. – Moskva : DMK Press, 2022. – 898 с.
2. Stivenson, B. Python : sbornik uprazhnenij / B. Stivenson ; perevod A. Ju. Gin'ko. – Moskva : DMK Press, 2021. – 238 с.
3. Uilks, M. Professional'naja razrabotka na Python / M. Uilks ; perevod A. A. Slinkin. – Moskva : DMK Press, 2021. – 502 с. – ISBN 978-5-97060-930-9.
4. Habarov, A. N. Razrabotka programnyh prilozhenij : uchebnik / A. N. Habarov, A. N. Ermakova. – Stavropol' : AGRUS, 2025.
5. Markin, A. V. Postreljacionnye bazy dannyh. MongoDB : uchebnoe posobie / A. V. Markin. – 3-e izd. – Moskva : Aj Pi Ar Media, 2024. – 383 с.
6. Volk, V. K. Bazy dannyh : uchebnik / V. K. Volk, V. Ju. Oseev, O. S. Cherepanov. – Moskva, Vologda : Infra-Inzhenerija, 2025. – 544 с.
7. Brjedshou, Sh. MongoDB: polnoe rukovodstvo. Moshhnaja i masshtabiruemaja sistema upravlenija bazami dannyh / Sh. Brjedshou, J. Brjezil, K. Hodorov ; perevod D. A. Belikov. – Moskva : DMK Press, 2020. – 540 с.
8. Bjenker, K. MongoDB v dejstvii / K. Bjenker ; perevod A. A. Slinkin. – 2-e izd. – Moskva : DMK Press, 2023. – 394 с.
9. Zuev, S. V. Metody analiza dannyh : uchebnoe posobie / S. V. Zuev. – Moskva : Aj Pi Ar Media, 2023. – 132 с.
10. Naumov, V. N. Analiz dannyh i mashinnoe obuchenie. Metody i instrumental'nye sredstva : uchebnoe posobie / V. N. Naumov. – Sankt-Peterburg : Severo-Zapadnyj institut upravlenija RANHiGS, 2020. – 260 с.

11. Zapechnikov, S. V. Osnovy intellektual'nogo analiza dannyh i mashinnogo obuchenija: konspekt lekcij : uchebnoe posobie / S. V. Zapechnikov. – Moskva : Nacional'nyj issledovatel'skij jadernyj universitet «MIFI», 2022. – 136 c.
12. Gorbachenko, V. I. Mashinnoe obuchenie: nastraivaem PO, gotovim dannye, analiziruem / V. I. Gorbachenko, K. E. Savenkov, M. A. Malahov. – Moskva, Almaty : Aj Pi Ar Media, EDP Hub (Idipi Hab), 2024. – 248 c.
13. Belov V.V. Proektirovanie informacionnyh sistem : uchebnik / Belov V.V., Chistjakova V.I.. – Moskva : KURS, 2024. – 400 c.
14. Grinchenko N.N. Proektirovanie informacionnyh sistem : uchebnik / Grinchenko N.N., Gromov A.Ju., Hizrieva N.I.. – Moskva : KURS, 2024. – 176 c.
15. Ciperman, G. N. Proektirovanie informacionnyh sistem : praktikum / G. N. Ciperman, N. A. Akatova. – Moskva : Izdatel'skij Dom MISiS, 2024. – 200 c.
16. Aleksandrovskaja, Ju. P. Informacionnye tehnologii statisticheskogo analiza dannyh : uchebno-metodicheskoe posobie / Ju. P. Aleksandrovskaja. – Kazan' : Kazanskij nacional'nyj issledovatel'skij tehnologicheskij universitet, 2019. – 152 c.
17. Djaminova, Je. I. Statisticheskij analiz dannyh s pomoshh'ju programmnyh sredstv : praktikum / Je. I. Djaminova, L. N. Titova, A. S. Filippova. – Saratov : Vuzovskoe obrazovanie, 2022. – 98 c.
18. Kubekov B.S. Modeli sistemy v inzhenerii programmnoho obespechenija : uchebnoe posobie / Kubekov B.S.. – Almaty, Moskva : EDP Hub (Idipi Hab), Aj Pi Ar Media, 2026. – 359 c.
19. Afonin, D. N. Tehnologii bol'shih dannyh : uchebnik / D. N. Afonin, P. N. Afonin. – Sankt-Peterburg : Intermedija, 2026. – 104 c.
20. Bystrenina, I. E. Proektirovanie informacionnyh sistem : uchebnoe posobie dlja vuzov / I. E. Bystrenina. – Moskva : Dashkov i K, 2026. – 118 c.
21. Martynov, A. P. Informacionnaja bezopasnost' i zashhita informacii : uchebnoe posobie / A. P. Martynov, I. A. Martynova, A. A. Rusakov. – 2-e izd. – Moskva : Aj Pi Ar Media, 2026. – 130 c.
22. Melikov, P. I. Izuchaem osnovy Python. Prakticheskij kurs dlja data-analitikov / P. I. Melikov. – Moskva, Almaty : Aj Pi Ar Media, EDP Hub (Idipi Hab), 2026. – 480 c.
23. Shustrov, A. S. Osnovy prikladnogo programmirovaniya na jazyke Python : uchebnoe posobie / A. S. Shustrov. – Moskva : Aj Pi Ar Media, 2026. – 91 c.
24. Apple Inc. Apple Health – platforma monitoringa zdorov'ja [Jelektronnyj resurs]. – Rezhim dostupa: <https://www.apple.com/health/> (data obrashhenija: 15.03.2025).
25. Google LLC. Google Fit – platforma monitoringa fizicheskoy aktivnosti [Jelektronnyj resurs]. – Rezhim dostupa: <https://www.google.com/fit/> (data obrashhenija: 15.03.2025).
26. Garmin Ltd. Garmin Connect – platforma analiza sportivnyh pokazatelej [Jelektronnyj resurs]. – Rezhim dostupa: <https://connect.garmin.com/> (data obrashhenija: 15.03.2025).
27. MongoDB Inc. MongoDB 7.0 Documentation – dokumentacija sistemy upravlenija bazami dannyh [Jelektronnyj resurs]. – Rezhim dostupa: <https://www.mongodb.com/docs/> (data obrashhenija: 15.03.2025).
28. The Qt Company. PySide6 Documentation – dokumentacija frejmvorka razrabotki graficheskikh prilozhenij [Jelektronnyj resurs]. – Rezhim dostupa: <https://doc.qt.io/qtforpython/> (data obrashhenija: 15.03.2025).
29. Statista GmbH. Wearable Technology – statistika mirovogo rynka nosimyh ustroystv [Jelektronnyj resurs]. – Rezhim dostupa: <https://www.statista.com/topics/1556/wearable-technology/> (data obrashhenija: 15.03.2025).

30. Groq Inc. Groq API Documentation – dokumentacija programmnogo interfejsa dlja raboty s jazykovymi modeljami [Elektronnyj resurs]. – Rezhim dostupa: <https://console.groq.com/docs/> (data obrashhenija: 15.03.2025).