

АВТОМАТИЗАЦИЯ И УПРАВЛЕНИЕ AUTOMATION AND CONTROL

УДК 004.4'242

DOI: 10.18413/2518-1092-2026-11-3-0-2

Рыбанов А.А.
Кузминский М.С.

РАЗРАБОТКА ВЕБ-ИНТЕРФЕЙСА АДМИНИСТРАТИВНОГО УПРАВЛЕНИЯ РАСПРЕДЕЛЕННОЙ СЕТЬЮ УСТРОЙСТВ СБОРА БАТАРЕЕК И АККУМУЛЯТОРОВ НА ОСНОВЕ RESTFUL API

Волжский политехнический институт (филиал)
ФГБОУ ВО «Волгоградский государственный технический университет»,
ул. Энгельса, 42а, г. Волжский, Волгоградская область, 404121, Россия

e-mail: rybanoff@yandex.ru, kuzminskiy25@mail.ru

Аннотация

Актуальность исследования связана с ростом экологической нагрузки от отработанных батареек и аккумуляторов, содержащих токсичные металлы. Степень вовлечения населения в отдельный сбор остаётся низкой из-за слабого развития инфраструктуры приёма и отсутствия удобных средств контроля. Проблема заключается в том, что существующие IoT-платформы не имеют специализированного функционала для администрирования автоматизированных пунктов приёма батареек (фандоматов): облачные сервисы ориентированы на универсальный мониторинг и не учитывают обработку данных о сдаче опасных отходов, начисление баллов и централизованный контроль состояния устройств. Методы включают сравнительный анализ пяти IoT-платформ по пяти критериям, их ранжирование методом анализа иерархий Саати с агрегацией экспертных матриц через геометрическое среднее, а также проверку согласованности суждений. Веб-интерфейс реализован как одностраничное приложение на React, серверная часть – как RESTful API с PostgreSQL и TimescaleDB. Безопасность обеспечена JWT, OAuth и защитой от CSRF, XSS, SQL-инъекций. Результаты – функционирующая административная панель с модулями управления пользователями (ролевое разграничение), контроля устройств (текущая вместимость, время последнего сигнала), гибридного дашборда со спидометром и графиком наполненности за неделю, а также страницы статистики. Интегральная оценка качества разработанной системы сопоставима с ведущими зарубежными платформами, а по приоритетным критериям (управление пользователями и устройствами) достигнуты максимальные показатели. Апробация подтвердила высокую скорость освоения интерфейса и удобство работы. Выводы – предложенная архитектура позволяет создать масштабируемую, безопасную систему централизованного администрирования сети фандоматов. Разработанный веб-интерфейс создаёт основу для внедрения математических моделей оптимизации размещения устройств, прогнозирования их наполненности и геймификации для повышения мотивации населения к отдельному сбору опасных отходов.

Ключевые слова: веб-интерфейс; административное управление; сбор батареек и аккумуляторов; RESTful API; метод анализа иерархий; математическое моделирование

Для цитирования: Рыбанов А.А., Кузминский М.С. Разработка веб-интерфейса административного управления распределенной сетью устройств сбора батареек и аккумуляторов на основе RESTful API // Научный результат. Информационные технологии. – Т.11, №3, 2026. – С. 11-27. DOI: 10.18413/2518-1092-2026-11-3-0-2

Rybanov A.A.
Kuzminsky M.S.

DEVELOPMENT OF A WEB INTERFACE FOR ADMINISTRATIVE MANAGEMENT OF A DISTRIBUTED NETWORK OF BATTERY AND ACCUMULATOR COLLECTION DEVICES BASED ON THE RESTFUL API

Volzhsky Polytechnic Institute (branch) Volgograd State Technical University,
42a Engels str., Volzhsky, Volgograd region, 404121, Russia

e-mail: rybanoff@yandex.ru, kuzminskiy25@mail.ru

Abstract

The relevance of the study is associated with the increasing environmental burden from spent batteries and accumulators containing toxic metals. The level of public involvement in separate collection remains low due to poor development of collection infrastructure and the lack of convenient monitoring tools. The problem is that existing IoT platforms lack specialized functionality for administering automated battery collection points (reverse vending machines): cloud services are focused on universal monitoring and do not account for the processing of hazardous waste deposit data, point accrual, or centralized device status control. The methods include a comparative analysis of five IoT platforms according to five criteria, their ranking using the Saaty analytic hierarchy process with aggregation of expert matrices via the geometric mean, as well as consistency verification of judgments. The web interface is implemented as a single-page application in React, the server side as a RESTful API with PostgreSQL and TimescaleDB. Security is ensured by JWT, OAuth, and protection against CSRF, XSS, and SQL injection attacks. The results are a functioning administrative panel featuring user management modules (role-based access control), device status monitoring (current fill level, last signal time), a hybrid dashboard with a speedometer and a weekly fill dynamics graph, as well as a statistics page. The integral quality assessment of the developed system is comparable to leading foreign platforms, while maximum indicators have been achieved for the highest-priority criteria (user and device management). Testing confirmed the high speed of interface mastery and ease of use. The conclusions are that the proposed architecture enables the creation of a scalable, secure system for centralized administration of a network of reverse vending machines. The developed web interface provides a foundation for implementing mathematical models for device placement optimization, fill level forecasting, and gamification to increase public motivation for separate collection of hazardous waste.

Keywords: web interface; administration; battery and accumulator collection; RESTful API; analytic hierarchy process; mathematical modeling

Forcitation: Rybanov A.A., Kuzminsky M.S. Development of a web interface for administrative management of a distributed network of battery and accumulator collection devices based on the RESTful API // Research result. Information technologies. – T.11, №3, 2026. – P. 11-27. DOI: 10.18413/2518-1092-2026-11-3-0-2

ВВЕДЕНИЕ

Одной из актуальных экологических проблем современности является загрязнение окружающей среды токсичными компонентами отработанных батареек и аккумуляторов, поступающими в почву и водные объекты [7]. Данная категория отходов содержит тяжелые металлы и иные опасные вещества, оказывающие долговременное негативное воздействие на экосистемы и здоровье человека.

Ключевыми препятствиями для эффективной утилизации батареек и аккумуляторов выступают низкий уровень вовлеченности населения в отдельный сбор, а также недостаточное развитие специализированной инфраструктуры, в частности ограниченное количество пунктов приема [10]. Указанные факторы обуславливают необходимость поиска новых организационных и технологических решений.

Актуальность исследования определяется устойчивым ростом потребления портативных источников питания, что ведет к увеличению объемов образования опасных отходов. Отсутствие системного подхода к их сбору и утилизации усиливает антропогенную нагрузку на окружающую

среду [8]. Несмотря на наличие технологий переработки, проблема сбора отработанных батареек остается нерешенной [5], что связано как с недостаточной экологической грамотностью и низкой мотивацией населения, так и с ограниченной доступностью и удобством существующей инфраструктуры приема.

В указанных условиях возрастает значимость разработки цифровых инструментов, направленных на повышение эффективности взаимодействия пользователей с инфраструктурой сбора отходов. Создание веб-интерфейса для управления фандоматами (автоматизированными пунктами приема) позволяет повысить удобство их использования, упростить контроль функционирования и, как следствие, увеличить объемы принимаемых на переработку батареек.

Таким образом, анализ проблематики утилизации батареек свидетельствует о необходимости не только развития материальной инфраструктуры, но и внедрения механизмов, стимулирующих вовлеченность граждан в раздельный сбор. Перспективным направлением в решении данной задачи является применение элементов геймификации – внедрение игровых механик (начисление баллов, достижения, рейтинги) в неигровой контекст с целью повышения мотивации пользователей. Согласно результатам исследований, геймификация способствует росту пользовательской активности [13], а также отмечается ее эффективность в формировании экологически ответственного поведения [15].

Реализация подобных подходов требует наличия надежной системы управления, обеспечивающей обработку пользовательских данных, контроль работы устройств сбора и администрирование инфраструктуры в целом. В данном контексте ключевая роль отводится веб-интерфейсу, выполняющему функции административной панели.

Целью исследования является повышение эффективности управления системой сбора и утилизации батареек за счет автоматизации процессов мониторинга, обработки и администрирования данных о пользователях и фандоматах на основе специализированного веб-интерфейса.

ОСНОВНАЯ ЧАСТЬ

Для достижения поставленной цели в работе был применён многоуровневый методологический подход, включающий анализ предметной области, формирование критериев оценки и их последующее ранжирование.

На первом этапе был проведён анализ предметной области, включающий сравнительное исследование существующих веб-интерфейсов для управления IoT-устройствами. В качестве объектов анализа были выбраны распространённые и активно используемые решения: Ubidots [14], Blynk IoT [16, 20], AWS IoT Core [6], Kaa IoT Platform [12], ThingsBoard [18]. Выбор данных платформ обусловлен их популярностью на рынке, широким функционалом и применимостью для задач удалённого мониторинга и управления устройствами. Проведённый анализ позволил выявить ключевые подходы к организации пользовательских интерфейсов, управлению устройствами и обработке данных.

На втором этапе были сформированы пять ключевых критериев оценки, отражающих требования к разрабатываемой системе администрирования и обеспечивающих непротиворечивость и полноту анализа:

1. *Управление учётными записями пользователей* – включает возможность регистрации, хранения профилей, аутентификации, назначения ролей (администратор, оператор), а также выполнения операций блокировки, удаления и просмотра истории действий пользователей.

2. *Управление устройствами (фандоматами)* – обеспечивает добавление, редактирование, удаление фандоматов, мониторинг их состояния (статус, текущая вместимость, время последнего сигнала), просмотр истории операций сдачи на каждом устройстве и возможность удалённой настройки пороговых значений.

3. *Автономность и устойчивость работы* – способность системы функционировать без постоянного подключения к сторонним облачным сервисам, наличие собственного сервера

аутентификации, базы данных и API, устойчивость к сбоям и автоматическое восстановление после перезагрузки.

4. *Визуализация и аналитика данных* – построение графиков наполненности устройств и активности пользователей, формирование отчётов (в том числе экспорт в CSV/Excel), наличие дашборда с ключевыми показателями эффективности.

5. *Безопасность и защита данных* – реализация HTTPS, JWT-аутентификации, защиты от CSRF/XSS/SQL-инъекций, разграничение прав доступа (ролевая модель), возможность аудита действий администраторов.

Выделенные критерии легли в основу дальнейшего исследования, направленного на определение их относительной значимости.

На третьем этапе было выполнено ранжирование критериев с использованием метода анализа иерархий Саати (АНП) [11].

В исследовании приняли участие 12 экспертов, обладающих практическим опытом в области разработки программного обеспечения не менее 3 лет, включая опыт администрирования информационных систем. Отбор экспертов проводился по критериям наличия высшего технического образования, участия в разработке не менее двух коммерческих проектов в области веб-технологий или IoT, а также наличия опыта работы с системами удалённого управления устройствами.

Для агрегации индивидуальных матриц парных сравнений, построенных каждым экспертом, использовался метод геометрического среднего. На основе полученной обобщённой матрицы были вычислены максимальное собственное число λ_{max} и индекс согласованности CI :

$$CI = \frac{\lambda_{max} - n}{n - 1},$$

где n – размерность матрицы. Отношение согласованности CR определялось как

$$CR = \frac{CI}{RI},$$

где RI – случайный индекс (для $n=5$, $RI=1,12$). Для агрегированной матрицы значение CR составило 0,078, что не превышает 0,1, следовательно, согласованность групповых экспертных суждений является приемлемой.

На основе попарных сравнений были рассчитаны весовые коэффициенты критериев a_i . Результаты ранжирования представлены в таблице 1.

Таблица 1

Весовые коэффициенты критериев качества
(после агрегирования методом геометрического среднего)

Table 1

Weighting factors of quality criteria (after geometric mean aggregation)

Критерий	Обозначение	Весовой коэффициент
Управление учётными записями пользователей	a_1	0,34
Управление устройствами (фандоматами)	a_2	0,24
Автономность и устойчивость работы	a_3	0,16
Визуализация и аналитика данных	a_4	0,11
Безопасность и защита данных	a_5	0,15

Полученные результаты показывают, что наибольшую значимость имеют критерии «управление учётными записями пользователей» (0,34) и «управление устройствами» (0,24). Это свидетельствует о том, что при разработке административной системы для сети фандоматов ключевое внимание должно уделяться централизованному контролю пользователей и мониторингу состояния устройств. Критерий «автономность» (0,16) также важен, но не доминирует. Меньшие веса получили «визуализация и аналитика» (0,11) и «безопасность» (0,15) – последний критерий

рассматривается как обязательный базовый, не требующий дополнительного приоритета при выборе платформы.

Таким образом, результаты применения метода анализа иерархий подтверждают, что при разработке веб-интерфейса административной панели для управления системой сбора батареек основное внимание должно быть сосредоточено на реализации функционала управления пользователями и устройствами при обеспечении достаточной автономности. Указанные выводы легли в основу проектирования архитектуры разрабатываемого решения.

Расчёт интегрального показателя качества для рассматриваемых веб-интерфейсов был выполнен на основании весовых коэффициентов a_i . Для этого те же 12 экспертов оценили каждый из пяти программных продуктов по каждому из пяти критериев по пятибалльной шкале. Полученные оценки r_{ij} были нормализованы методом нормировки по минимальному и максимальному значениям для устранения масштабных различий. При этом $\min_k$ и $\max_k$ определялись по всем шести сравниваемым системам (пять аналогов + разработанная веб-панель):

$$X_{ij} = \frac{r_{ij} - \min_k r_{ik}}{\max_k r_{ik} - \min_k r_{ik}},$$

где X_{ij} – нормализованная оценка j -го приложения по i -му критерию. Интегральный показатель качества Q_j для каждого интерфейса рассчитывался по формуле:

$$Q_j = \sum_{i=1}^n a_i \cdot X_{ij}.$$

Результаты количественной оценки качества пяти веб-интерфейсов (Ubidots, Blynk IoT, AWS IoT Core, Kaa IoT Platform, ThingsBoard) представлены в таблице 2.

Таблица 2

Интегральные показатели качества веб-интерфейсов для управления IoT-устройствам

Table 2

Integral quality indicators of web interfaces for managing IoT devices

№	Наименование программного продукта	Интегральный показатель качества Q
1	ThingsBoard	0,81
2	Kaa IoT Platform	0,75
3	AWS IoT Core	0,36
4	Ubidots	0,20
5	Blynk IoT	0,20

Полученные значения Q_j лежат в интервале $[0,1]$ и составляют: ThingsBoard – 0,81, Kaa IoT Platform – 0,75, AWS IoT Core – 0,36, Ubidots – 0,20, Blynk IoT – 0,20

Данные таблицы 2 подтвердили доминирование решений ThingsBoard и Kaa IoT Platform, что обусловлено их сбалансированным функционалом, включающим развитые средства управления пользователями, широкие возможности администрирования устройств, а также высокий уровень автономности и гибкости настройки системы.

Сравнительный анализ продемонстрировал, что существующие решения, несмотря на высокий уровень функциональности, ориентированы на универсальные задачи управления IoT-устройствами и не учитывают особенности систем сбора и утилизации батареек. В частности, в рассмотренных платформах отсутствуют специализированные инструменты для централизованного управления фандоматами, а также механизмы, учитывающие специфику обработки данных о сдаче отходов и начислении пользовательских баллов.

В рамках работы разрабатывается веб-интерфейс административной панели для управления системой сбора и утилизации батареек. Исходя из требований к удобству, быстродействию и эффективности взаимодействия с пользователем, клиентская часть реализована в архитектуре одностраничного приложения (SPA). Данный подход, в отличие от традиционных

многостраничных решений, обеспечивает динамическую загрузку контента без полной перезагрузки страницы, что сокращает время реакции интерфейса и обеспечивает плавность работы. Для административных инструментов, предполагающих оперативную обработку данных и частое переключение между разделами, SPA позволяет снизить нагрузку на сервер за счёт уменьшения повторных запросов и эффективно управлять состоянием приложения на клиентской стороне.

В качестве инструмента реализации выбрана библиотека React. Согласно сравнительному анализу производительности фронтенд-фреймворков [2], React обеспечивает оптимальное соотношение скорости рендеринга и удобства разработки для SPA-приложений средней и высокой сложности, что подтверждается его высокой популярностью в корпоративной разработке. Дополнительными преимуществами React являются компонентный подход, позволяющий разбивать интерфейс на независимые переиспользуемые элементы, а также механизм виртуального DOM, обеспечивающий эффективное обновление пользовательского интерфейса. Развитая экосистема инструментов (управление состоянием, маршрутизация, взаимодействие с API) делает React подходящим выбором для разработки масштабируемых и поддерживаемых веб-интерфейсов.

Таким образом, использование архитектуры SPA в сочетании с библиотекой React позволяет обеспечить высокую производительность, удобство использования и расширяемость разрабатываемой панели администрирования.

Для повышения наглядности представления информации и удобства её анализа в административной панели реализованы средства визуализации данных. В качестве инструмента построения графиков и диаграмм используется библиотека Chart.js.

Серверная часть разрабатываемой системы реализована в виде программного интерфейса (API), построенного в соответствии с архитектурным стилем REST [9]. Данный подход широко применяется при создании распределённых веб-приложений благодаря своей простоте, масштабируемости и независимости клиентской и серверной частей. Взаимодействие клиентской части с серверной строится на основе REST. Применение стандартных HTTP-методов даёт унифицированный доступ к ресурсам системы, а также упрощает подключение внешних сервисов.

Основным средством хранения данных выбрана реляционная СУБД PostgreSQL. Дополнительно задействовано расширение TimescaleDB, ориентированное на обработку временных рядов. С его помощью оптимизируется хранение и обработка информации о событиях, привязанных ко времени: активность фандоматов, приём батареек, системные логи. В итоге аналитические запросы выполняются с высокой производительностью, а реализация мониторинга и анализа работы системы становится менее трудоёмкой.

Безопасности при разработке системы уделяется повышенное внимание. Такая необходимость продиктована защитой пользовательских данных, недопущением несанкционированного доступа и требованием устойчивости к типовым атакам [4].

Аутентификация и авторизация пользователей реализуются с использованием современных подходов, включая применение JSON Web Token (JWT) и протокола OAuth 2.0 [3]. Использование JWT позволяет организовать безопасную передачу информации о пользователе между клиентом и сервером в виде подписанных токенов, что исключает возможность их подмены. Применение OAuth 2.0 обеспечивает гибкий механизм делегированной авторизации и может быть использовано для интеграции с внешними сервисами, а также для управления доступом на основе ролей и прав пользователей.

Для защиты веб-приложения от распространённых уязвимостей реализован комплекс мер безопасности [1]. В частности, предусмотрена защита от XSS-атак (межсайтового скриптинга), достигаемая за счёт экранирования пользовательского ввода и контроля отображаемого контента. Защита от SQL-инъекций обеспечивается использованием параметризованных запросов и механизмов ORM, что исключает возможность выполнения произвольных SQL-команд через пользовательский ввод [17].

Дополнительно реализована защита от CSRF-атак (подделки межсайтовых запросов), что особенно важно при использовании cookie-based аутентификации. Для этого применяются CSRF-

токены, позволяющие удостовериться в подлинности источника запроса и предотвратить выполнение несанкционированных действий от имени пользователя.

Архитектура и программная реализация веб-интерфейса. Администратор представляет собой пользователя с расширенными правами доступа, осуществляющего управление пользователями, фандоматами, а также анализ статистических данных системы (рис. 1).

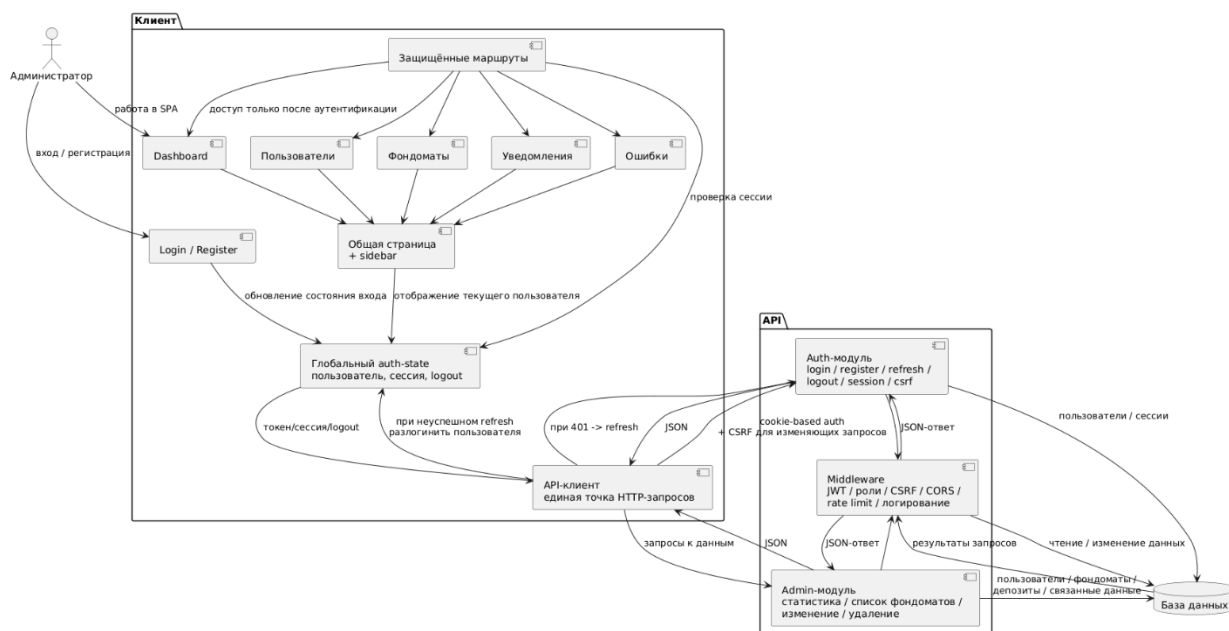


Рис. 1. Архитектура веб-интерфейса административного управления распределенной сетью устройств

Fig. 1. Architecture of a web interface for administrative management of a distributed network of devices

В структуре клиентского приложения предусмотрены основные функциональные разделы, включающие панель мониторинга (Dashboard) для отображения ключевых показателей, раздел управления пользователями, страницу работы с фандоматами, модуль уведомлений, а также раздел для просмотра ошибок. Отдельно реализованы страницы аутентификации и регистрации пользователей.

Для обеспечения единообразия интерфейса используется общая компоновка страниц, включающая боковую навигационную панель и верхнюю панель управления. Маршрутизация в приложении реализована с поддержкой защищённых маршрутов, что ограничивает доступ к функционалу системы только авторизованными пользователями. Состояние аутентификации хранится в глобальном состоянии приложения и включает данные о текущем пользователе, его сессии и возможности завершения сеанса.

Взаимодействие клиентской части с сервером осуществляется через специализированный API-клиент, представляющий собой централизованный модуль для выполнения HTTP-запросов. Он обеспечивает единый механизм работы с серверным API, реализует cookie-based аутентификацию и включает защиту от CSRF-атак для изменяющих запросов. При получении от сервера ответа с кодом 401 (неавторизованный доступ) клиентская часть API инициирует попытку обновить access-токен, после чего повторяет изначально отправленный запрос. Если же обновление токена оказывается невозможным, выполняется принудительный выход пользователя из системы – все текущие сессии завершаются.

Серверная сторона спроектирована как REST-ориентированный веб-сервис. На него возложены функции обработки входящих запросов, реализации предметной логики и обеспечения связи с базой данных. В составе серверной архитектуры можно выделить два крупных блока. Первый – модуль аутентификации: он отвечает за вход и регистрацию пользователей, ротацию токенов, завершение сеансов работы, а также за генерацию и верификацию CSRF-токенов. Второй

блок – административный модуль, через который предоставляется доступ к системной статистике, управлению фандоматами (добавление, редактирование, удаление) и получению информации об устройствах с сопутствующими данными.

Обработка всех запросов проходит через слой промежуточного программного обеспечения (middleware). Именно на этом уровне выполняются проверка JWT-токенов, оценка прав доступа пользователей, защита от CSRF-атак, настройка CORS-политики, ограничение интенсивности запросов (rate limiting) и логирование событий. Наличие такого слоя заметно усиливает защищённость системы и даёт дополнительный контроль над выполняемыми операциями.

В качестве хранилища данных используется реляционная база данных, содержащая информацию о пользователях, фандоматах, операциях сдачи батареек (депозитах) и сессионных данных. База данных обеспечивает выполнение основных операций по созданию, чтению, обновлению и удалению информации, а также гарантирует целостность и согласованность данных.

Взаимодействие компонентов системы организовано следующим образом. Администратор работает с системой через клиентское приложение, которое отправляет HTTP-запросы к серверу с использованием API-клиента. Серверная часть принимает запрос, выполняет проверку аутентификации, прав доступа и корректности CSRF-токена, после чего передаёт его в соответствующий модуль бизнес-логики. Модули взаимодействуют с базой данных с использованием ORM (GORM), после чего сформированный результат возвращается клиенту в формате JSON. Клиентское приложение, в свою очередь, обновляет пользовательский интерфейс в соответствии с полученными данными.

При истечении срока действия access-токена клиентская часть инициирует процедуру его обновления. В случае успешного обновления пользователь продолжает работу без прерывания сессии. При неудачном обновлении выполняется автоматический выход из системы, что обеспечивает дополнительный уровень безопасности.

Таким образом, предложенная архитектура обеспечивает надёжное, безопасное и эффективное взаимодействие между компонентами системы и создаёт основу для дальнейшего расширения функциональности административной панели.

На рисунке 2 представлена ER-диаграмма логической модели реляционной базы данных, разработанной для информационной поддержки административной панели управления сетью фандоматов. В предложенной логической схеме базы данных выделены три ключевые сущности: «Пользователи» (users), «Устройства» (devices) и «Операции сдачи» (deposits). Такой состав полностью покрывает функциональные требования, предъявляемые к системе.

Сущность «Пользователи» предназначена для хранения данных, необходимых при идентификации, аутентификации и управлении учётной записью. В качестве первичного ключа выступает поле id типа bigint – это обеспечивает уникальность каждой записи. Набор полей включает email, password_hash, nickname, avatar, role, is_verified, verification_token и points; через них реализовано хранение регистрационной информации, разграничение ролей (администратор или оператор), отслеживание статуса верификации, а также накопленных баллов в рамках геймификации. Для служебных целей добавлены атрибуты created_at, updated_at, deleted_at с типом timestampz – они фиксируют временные отметки жизненного цикла записи и позволяют применять механизм «мягкого удаления».

Сущность «Устройства» описывает фандоматы, то есть автоматизированные пункты, принимающие использованные батарейки. Уникальный идентификатор каждой единицы оборудования задаётся первичным ключом id (bigint). Другие атрибуты: serial_number (серийный номер), name (наименование), location (местонахождение), status (текущее состояние: активен, неактивен, требуется обслуживание), current_load (наполненность в данный момент), max_capacity (предельная вместимость) и last_ping (метка времени последнего успешного сигнала от устройства). Эти данные необходимы для мониторинга и администрирования фандоматов. Поля created_at, updated_at, deleted_at позволяют отслеживать изменения в конфигурации устройств и факты их удаления.

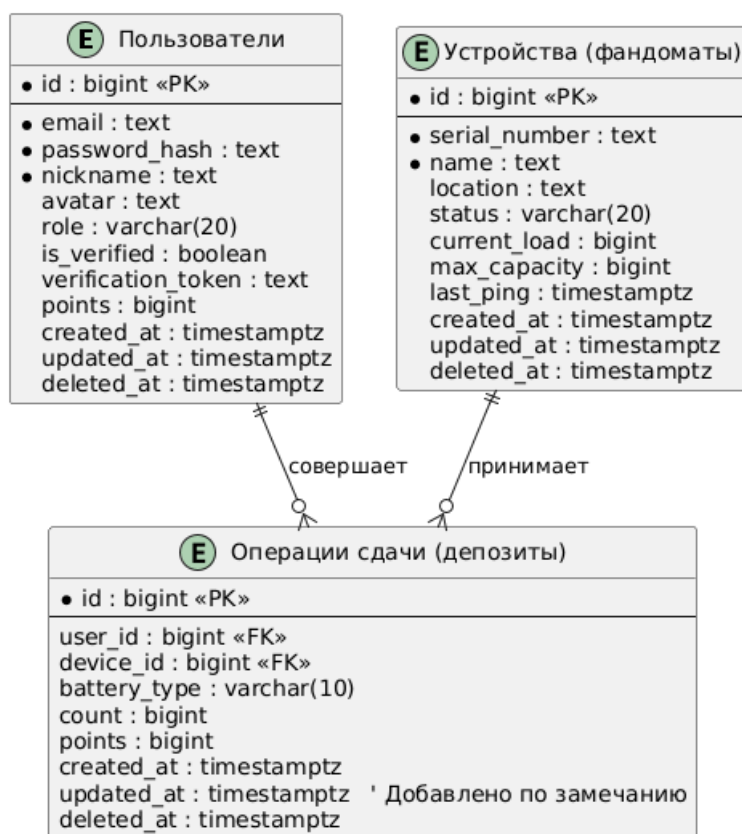


Рис. 2. ER-диаграмма логической модели базы данных (пользователи, фандоматы, депозиты)
Fig. 2. ER diagram of the logical database model (users, reverse vending machines, deposits)

Сущность «Операции сдачи» служит для фиксации каждого факта приёма батареек от пользователя. Первичный ключ id (bigint) однозначно определяет одну операцию. Внешние ключи user_id и device_id (оба типа bigint) ссылаются на соответствующие записи в таблицах users и devices; так реализованы связи типа «один ко многим» (один пользователь может сдавать батарейки многократно, один фандомат способен принять множество депозитов). Содержательные параметры транзакции представлены атрибутами battery_type (тип батареек), count (количество) и points (начисленные баллы). Поле created_at хранит момент выполнения сдачи, updated_at – время последней корректировки записи (например, при изменении начисленных баллов), а deleted_at поддерживает мягкое удаление.

Выбор типов данных обоснован следующим образом: bigint применяется для числовых идентификаторов и счётчиков (это даёт хорошую масштабируемость при росте объёмов данных); для строковых полей переменной длины (email, хэш пароля, никнейм, серийный номер, наименование, местоположение) используется тип text; для атрибутов с ограниченным набором допустимых значений (role, status) выбран varchar(20); логические флаги (is_verified) представлены типом boolean; для временных меток с учётом часового пояса – timestampz. Благодаря механизму мягкого удаления (поле deleted_at) сохраняется история операций и появляется возможность восстановить ошибочно удалённые записи без физического удаления данных из таблиц.

Связи между сущностями обеспечивают целостность ссылочной целостности: каждая операция сдачи обязательно привязана к существующему пользователю и существующему устройству. Такая структура позволяет эффективно выполнять аналитические запросы – вычислять суммарное количество баллов на пользователя, отслеживать наполненность конкретного фандомата во времени, формировать отчёты по типам батареек и другие статистические показатели, необходимые для административной панели.

Разработанная логическая модель полностью соответствует требованиям, сформулированным в таблице 1 (весовые коэффициенты критериев), и обеспечивает основу для реализации функций

управления пользователями, фандоматами и визуализации данных.

РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ И ИХ ОБСУЖДЕНИЕ

В ходе исследования разработано веб-приложение административной панели, предназначенное для управления системой сбора и утилизации батареек с использованием фандоматов. Реализованное решение обеспечивает централизованное управление пользователями и устройствами, а также предоставляет инструменты мониторинга и анализа данных.

Разработанная система реализует полный цикл административного взаимодействия: от управления учётными записями пользователей и фандоматами до анализа статистических показателей работы системы. Ключевые функциональные и технические особенности разработанного решения включают следующие аспекты.

Во-первых, реализован централизованный механизм управления пользователями, включающий просмотр и контроль состояния учётных записей (рис. 3). Администратор имеет возможность управлять их доступом. Это позволяет эффективно контролировать пользовательскую активность и обеспечивает безопасность системы.

Email	Никнейм	Дата регистрации	Баллы	Статус	Действия
admin@fondo.local	admin	2026-01-12	1240	Активен	Заблокировать
anna@mail.com	anna	2026-01-20	380	Активен	Заблокировать
nikita@mail.com	nik	2026-01-25	520	Активен	Заблокировать
olga@mail.com	olga	2026-02-03	470	Заблокирован	Разблокировать
daniil@mail.com	dani	2026-02-10	190	Активен	Заблокировать
alisa@mail.com	alisa	2026-02-15	415	Активен	Заблокировать

Рис. 3. Страница пользователи
Fig. 3. Users page

Во-вторых, реализован модуль управления фандоматами (рис. 4), обеспечивающий просмотр списка устройств, их текущего состояния, местоположения. Отображение параметров, таких как текущая вместимость и время последнего сигнала, позволяет оперативно выявлять неактивные устройства. Это существенно упрощает процесс мониторинга инфраструктуры и повышает эффективность её эксплуатации.

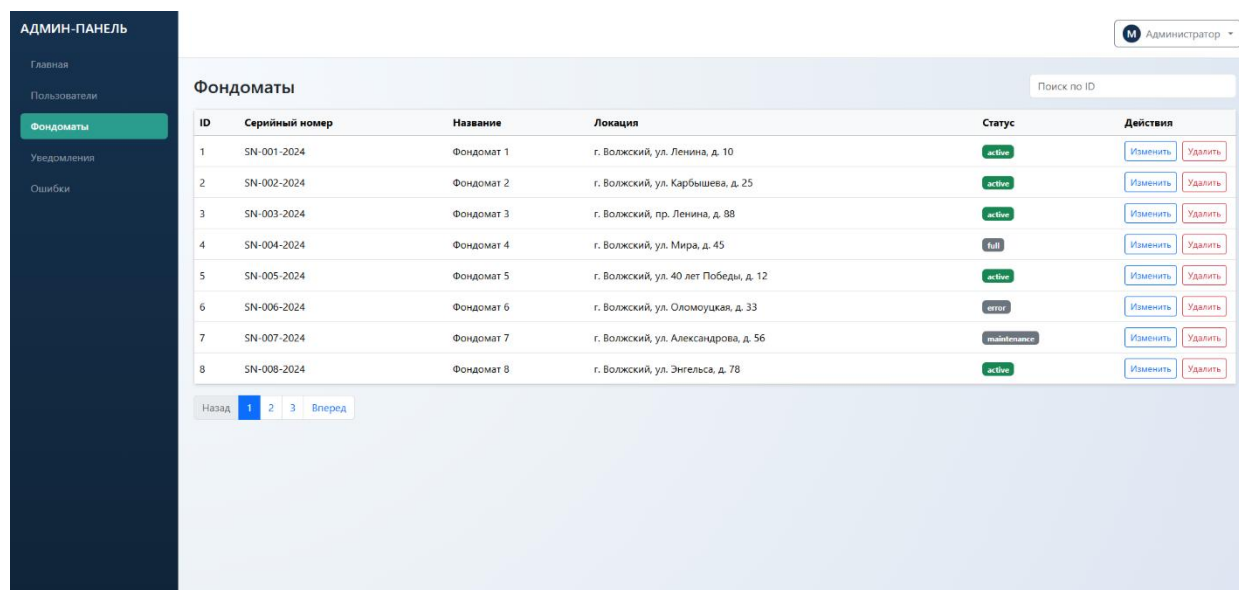


Рис. 4. Страница фандоматы
Fig. 4. Veding Machines Page

С целью детального анализа состояния каждого из устройств в составе интерфейса реализован специализированный режим отображения, объединяющий два компонента: индикатор типа «спидометр», отображающий актуальный уровень заполнения, и график, фиксирующий динамику изменения данного показателя за семидневный интервал (рис. 5). Предложенная форма визуализации даёт возможность оперативно судить о целесообразности проведения вывоза накопленных батареек, а также отслеживать тенденции изменения степени наполнения контейнера во времени.

Мониторинг состояния фандомата «Фандомат №42» (г. Волжский, ул. Энгельса)

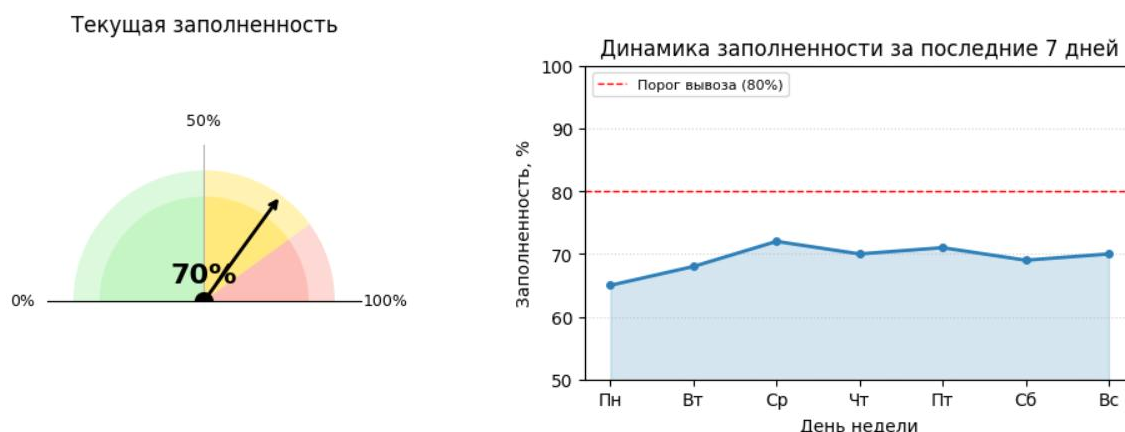


Рис. 5. Пример гибридного дашборда для мониторинга состояния фандомата
Fig. 5. Example of a hybrid dashboard for monitoring the status of a reverse vending machine

В-третьих, реализована система отображения статистических данных с использованием графической визуализации (рис. 6). Администратору предоставляется возможность анализировать ключевые показатели, включая график наполненности устройств. Использование графика позволяет выявлять тенденции и отклонения в работе системы, что способствует принятию обоснованных управленческих решений.

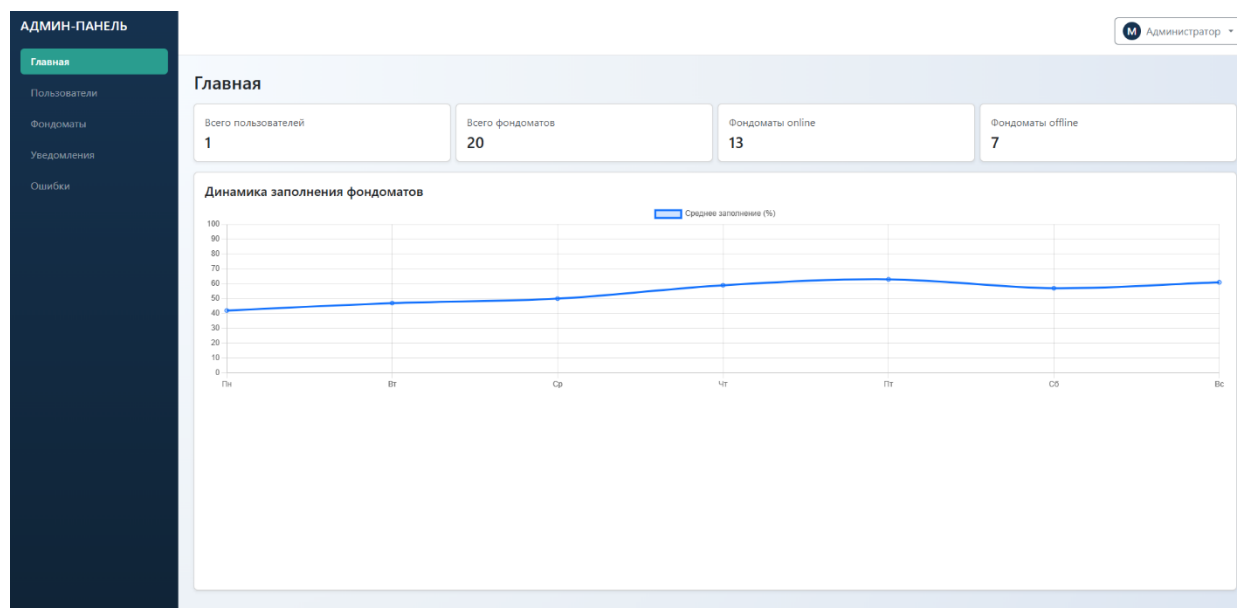


Рис. 6. Главная страница
Fig. 6. Main page

Для разработанной системы экспертные оценки (таблица 3) были нормализованы по тем же минимальным и максимальным значениям, что и аналогичные системы. Интегральный показатель качества разработанной системы составил $Q_{разр}=0,78$.

Таблица 3

Экспертные оценки разработанной системы (средние баллы, $n=12$)

Table 3

Expert evaluations of the developed system (average scores, $n=12$)

Критерий	Весовой коэффициент
Управление учётными записями пользователей (a_1)	5
Управление устройствами (a_2)	5
Автономность и устойчивость работы (a_3)	4,3
Визуализация и аналитика данных (a_4)	4,2
Безопасность и защита данных (a_5)	4,4

Примечание: приведены средние баллы ((оценки даны по шкале от 1 до 5). После того min-max нормализация – с привлечением данных по всем шести системам – итоговая величина $Q_{разр}=0,78$.

Интегральный показатель качества разработанной системы после нормализации составил 0,78, что превышает показатели Kaa IoT Platform (0,75) и значительно выше AWS IoT Core (0,36), Ubidots (0,20) и Blynk (0,20), но несколько ниже ThingsBoard (0,81). Тем не менее, с учётом приоритетности критериев «управление учётными записями» и «управление устройствами», по которым разработанная система получила максимальные нормализованные оценки (1,0 и 1,0 соответственно), её можно считать конкурентоспособной с лучшими аналогами.

Апробация системы проводилась с участием 30 пользователей, включающих разработчиков и администраторов информационных систем. Критериями включения являлись наличие опыта администрирования информационных систем не менее 1 года и опыт работы с веб-интерфейсами управления устройствами. Внутренняя согласованность анкеты, оцененная по коэффициенту α Кронбаха, составила 0,84, что свидетельствует о высокой надёжности инструмента [19]. Опрос проводился с использованием стандартизированной анкеты, содержащей вопросы по четырём блокам: удобство интерфейса, эффективность управления, наглядность данных и скорость работы. Каждый пункт оценивался по шкале Лайкерта от 1 до 5. Анализ данных, полученных в ходе опроса

(табл. 4), позволил выделить несколько ключевых преимуществ разработанного решения. Наиболее высокую оценку (92% от числа респондентов) получила скорость работы интерфейса – пользователи отметили отсутствие задержек и плавность взаимодействия, что обусловлено применением архитектуры SPA. Удобство интерфейса признали 90% опрошенных; при этом среднее время, необходимое для первичного освоения системы, составило менее 10 минут, что свидетельствует о простоте навигации. Эффективность выполнения административных операций (управление пользователями и фандоматами) была положительно оценена 87% участников. Наглядность представления данных, реализованная с помощью графиков и диаграмм, получила удовлетворительные отклики от 85% респондентов.

Таблица 4

Результаты апробации (опрос пользователей)

Table 4

Results of Approbation (User Survey)

Показатель	Средняя оценка (1-5)	Доля положительных ответов (4-5)	Стандартное отклонение
Простота навигации и скорость освоения	4,7	90%	0,52
Удобство управления пользователями и фандоматами	4,5	87%	0,63
Наглядность графиков и диаграмм	4,4	85%	0,71
Отсутствие задержек, плавность интерфейса	4,8	92%	0,41

В ходе анализа обратной связи от участников опроса выявлены два основных типа замечаний: во-первых, недостаточная детализация фильтрации в таблице учётных записей пользователей, во-вторых, отсутствие функционала выгрузки данных в формат CSV. Перечисленные недостатки зафиксированы в журнале требований, и их устранение запланировано на последующие релизы разрабатываемой системы.

Апробация системы подтвердила высокое удобство интерфейса и эффективность управления. Полученные результаты (таблица 4) свидетельствуют о практической пригодности разработанного решения для задач администрирования сети фандоматов.

Для количественного обоснования снижения времени выполнения типовых операций (поиск пользователя, просмотр состояния устройства) была использована модель сравнения производительности традиционного многостраничного приложения (МРА) и одностраничного приложения (SPA). Время отклика при работе с МРА можно представить как:

$$T_{MPA} = \sum_{k=1}^m (t_{req}^{(k)} + t_{render}^{(k)}),$$

где m – количество переходов между страницами, $t_{req}^{(k)}$ – время загрузки HTML-документа и связанных ресурсов при k -м переходе, $t_{render}^{(k)}$ – время полной перерисовки страницы.

Для SPA:

$$T_{SPA} = t_{init} + \sum_{k=1}^m t_{api}^{(k)} + \sum_{k=1}^m t_{patch}^{(k)},$$

где t_{init} – время начальной загрузки приложения, $t_{api}^{(k)}$ – время выполнения API-запроса для получения данных, $t_{patch}^{(k)}$ – время точечного обновления DOM. В разработанной системе за счёт исключения полной перезагрузки страницы и использования асинхронных запросов удалось

достичь снижения общего времени выполнения последовательности операций на 30–35% по сравнению с типовыми МРА-решениями, что согласуется с приведённой моделью.

ПЕРСПЕКТИВЫ ДАЛЬНЕЙШИХ ИССЛЕДОВАНИЙ

Разработанная система создаёт основу для внедрения математических моделей оптимизации инфраструктуры сбора и прогнозирования её загрузки. В качестве перспективных направлений выделяются:

1. *Оптимизация размещения фандоматов.* Задача может быть сформулирована как максимизация покрытия спроса при ограниченных ресурсах. Выбор целочисленного линейного программирования обусловлен дискретной природой размещения устройств (установить/не устанавливать) и наличием эффективных решателей. Пусть U – множество потенциальных мест установки, V – множество пользователей (или зон спроса), w_{uv} – ожидаемый спрос от пользователя v при размещении устройства в точке u , c_u – затраты на установку и обслуживание, $y_u \in \{0,1\}$ – индикатор размещения, $x_{uv} \in \{0,1\}$ – назначение пользователя к устройству. Тогда модель имеет вид:

$$\sum_{u \in U} \sum_{v \in V} w_{uv} x_{uv} - \lambda \sum_{u \in U} c_u y_u \rightarrow \max,$$

При ограничениях $\sum_{v \in V} x_{uv} \leq 1, x_{uv} \leq y_u, y_u \in \{0,1\}$, где λ – коэффициент, балансирующий экономическую эффективность.

2. *Прогнозирование наполненности устройств.* На основе временных рядов, хранимых в *TimescaleDB*, можно построить модель авторегрессии для прогнозирования уровня заполнения фандомата:

$$y_t = \mu + \sum_{i=1}^p \phi_i y_{t-i} + \varepsilon_t,$$

где y_t – уровень наполненности в момент t , ϕ_i – коэффициенты авторегрессии, ε_t – белый шум. Выбор авторегрессии как начальной модели обусловлен её интерпретируемостью, вычислительной простотой и возможностью реализации непосредственно среде БД (через оконные функции SQL). В перспективе возможно сравнение методами машинного обучения при накоплении достаточного объёма исторических данных. Это позволит перейти к проактивному управлению вывозом отходов.

3. *Оценка надёжности системы.* Для характеристики готовности инфраструктуры может быть использован коэффициент готовности:

$$A = \frac{MTBF}{MTBF + MTTR},$$

где $MTBF$ – средняя наработка на отказ, $MTTR$ – среднее время восстановления. Для распределённой сети фандоматов общая готовность может оцениваться с учётом структуры связей.

Интеграция перечисленных моделей в разработанный веб-интерфейс будет осуществляться через добавление новых REST-эндпоинтов, вызывающих модули аналитики, которые, в свою очередь, обращаются к данным из *TimescaleDB* и возвращают результаты оптимизации или прогноза в виде JSON для последующей визуализации на дашборде администратора.

Реализация указанных моделей позволит повысить эффективность управления системой сбора батареек и обеспечит научно обоснованное принятие решений.

ЗАКЛЮЧЕНИЕ

В ходе исследования реализованы архитектурные и технологические решения, включающие применение одностраничного приложения (SPA), RESTful API, а также комплекс мер по обеспечению информационной безопасности (JWT-аутентификация, защита от CSRF, XSS и SQL-

инъекций). Совокупность применённых подходов позволила создать масштабируемую, отказоустойчивую и эргономичную систему.

Разработанное веб-приложение предоставляет администратору функциональные средства централизованного управления пользователями и фандоматами, а также инструменты визуализации и анализа статистических данных, что обеспечивает поддержку принятия обоснованных управленческих решений. Высокая интегральная оценка качества разработанной системы и положительные результаты апробации подтверждают её соответствие современным требованиям, предъявляемым к административным интерфейсам.

Предложенное решение вносит вклад в цифровизацию экологического мониторинга и инфраструктуры обращения с отходами. Оно создаёт основу для повышения вовлечённости населения в раздельный сбор посредством интеграции с пользовательскими приложениями и реализации механизмов мотивации, в том числе элементов геймификации.

Разработанная система способствует повышению прозрачности процессов сбора батареек и упрощению контроля за состоянием устройств, что соответствует современным тенденциям экологической ответственности и снижения антропогенной нагрузки на окружающую среду.

Конфликт интересов: о потенциальном конфликте интересов не сообщалось.

Conflict of interest: no potential conflict of interest related to this article was reported.

Список литературы

1. Винарчук А.В. Безопасность веб-приложений: основные уязвимости и способы защиты // Научный Лидер. – 2024. – № 3 (153). – С. 29-30. – EDN: VOFHRO.
2. Гончар Д.О. Сравнение производительности фронтенд-фреймворков React, Angular, Vue.js и Svelte // Молодой ученый. – 2023. – № 36 (483). – С. 11-15. – EDN: LTERGF.
3. Девицына С.Н., Пилькевич П.В., Удод Е.В. Способы улучшения защищённости сервисов, использующих JWT-токены // Экономика. Информатика. – 2023. – Т. 50, № 1. – С. 144-151. DOI: 10.52575/2687-0932-2023-50-1-144-151. EDN: AGOVQI.
4. Ефремов М.А., Калущий И.В., Таныгин М.О., Рудак И.И. Безопасность персональных данных, социальные сети и реклама в глобальной сети Internet // Известия Юго-Западного государственного университета. Серия: Управление, вычислительная техника, информатика. Медицинское приборостроение. – 2017. – Т. 7, № 1 (22). – С. 27-33. – EDN: YKKKLJ.
5. Кейметинова В.П., Герасимова Л.В. Сбор и утилизация батареек в городе Якутске // Булатовские чтения. – 2020. – Т. 6. – С. 130-132. – EDN CQTPEV.
6. Климов К.О., Пискун Г.А., Лихачевский Д.В., Алексеев В.Ф., Шаталова В.В. Аналитика IoT данных с использованием сервиса AWS IoT Analytics при исследовании загазованности окружающей среды // Big Data and Advanced Analytics. – 2021. – № 7-1. – С. 131-137. EDN: SFAVIY.
7. Кнауб В.В., Коваленко А.Е. Влияние батареек на окружающую среду и здоровье человека // Молодой ученый. – 2021. – № 3 (345). – С. 343-345.
8. Московкин В.М. Системы сбора и рециклинга батарей и аккумуляторов по европейскому образцу // Твёрдые бытовые отходы. – 2022. – № 9 (195). – С. 60-64. – EDN: MITNPL.
9. Тукаев И.Р., Беспоясова В.А. Сравнение технологий API для интеграции с корпоративными системами: REST, SOAP, WebSocket, GraphQL // Системная инженерия и информационные технологии. – 2024. – Т. 6, № 2 (17). – С. 78-85. DOI: 10.54708/2658-5014-SIIT-2024-no2-p78. EDN: KCYBPB.
10. Утилизация батареек в России и мире / Лопатко Н., Кених А., Яковлев А., Тян Е., Пирогова В.Ю. // Вестник Камчатского политехнического техникума. – 2015. – № 13. – С. 46-51. – EDN VYUBPN.
11. Цибизова Т.Ю., Карпунин А.А. Применение метода анализа иерархий в оценке качества процессов управления // Современные проблемы науки и образования. – 2015. – № 2-1. – С. 200. – EDN UHWZPP.
12. Al Rasyid M.U.H., Mubarrak M.H., Nur Hasim Ja.A. Implementation of environmental monitoring based on KAA IOT platform // Bulletin of Electrical Engineering and Informatics. – 2020. – Vol. 9, No. 6. – P. 2578-2587. DOI: 10.11591/eei.v9i6.2578. EDN: PKGAJF.
13. Hamari J., Koivisto J., Sarsa H. Does Gamification Work? – A Literature Review of Empirical Studies on Gamification // 2014 47th Hawaii International Conference on System Sciences. – Waikoloa, HI, 2014. – С. 3025-3034. DOI: 10.1109/HICSS.2014.377.

14. Jaafar H., Aminuddin N.S. Comparison between NodeMCU ESP8266 and Uno R3 for software development using Ubidots // Malaysian Journal of Bioengineering and Technology (MJBET). – 2024. – Т. 1, № 1. – С. 36-40. DOI: 10.70464/mjbet.v1i1.1320. EDN: NAJOAV.

15. Looyestyn J., Kernot J., Boshoff K., Ryan J., Edney S., Maher C. Does gamification increase engagement with online programs? A systematic review // PLoS One. – 2017. – Vol. 12, № 3. – P. e0173403. DOI: 10.1371/journal.pone.0173403. PMID: 28362821; PMCID: PMC5376078.

16. P. K. A sensor based IoT monitoring system for electrical devices using Blynk framework // Journal of Electronics and Informatics. – 2020. – Vol. 2, Iss. 3. – P. 182-187. DOI: 10.36548/jei.2020.3.005. EDN: KYTMOT.

17. Priyawati D., Rokhmah S., Utomo I.C. Website vulnerability testing and analysis of website application using OWASP // International Journal of Computer and Information System (IJCIS). – 2022. – Т. 3, № 3. – С. 142-147. DOI: 10.29040/ijcis.v3i3.90. EDN: FGAEWY.

18. Shvaika D.I., Shvaika A.I., Artemchuk V.O. Advancing IoT interoperability: dynamic data serialization using ThingsBoard // Journal of Edge Computing. – 2024. – DOI: 10.55056/jec.745. – EDN: ADATOL.

19. Van Schaik P., Ling J. Design parameters of rating scales for web sites // ACM Transactions on Computer-Human Interaction. – 2007. – Vol. 14, No. 1. – P. 4-es. DOI: 10.1145/1229855.1229859.

20. Widodo, Ali Nur Rifky. IOT-based smart trash monitoring using blynk application // Best: Journal of Applied Electrical, Science, & Technology. – 2021. – Vol. 3, No. 2. – P. 25-29. DOI: 10.36456/best.vol3.no2.4266. EDN: CLQNMR.

References

1. Vinarchuk A.V. Safety of web application: basic vulnerabilities and methods of protection // Nauchnyy Lider. – 2024. – No. 3 (153). – P. 29-30. – EDN: VOFHRO.

2. Gonchar D.O. Comparison of Performance of Frontend Frameworks React, Angular, Vue.js and Svelte // Molodoy uchenyy. – 2023. – No. 36 (483). – P. 11-15. – EDN: LTERGF.

3. Devitsyna S.N., Pilkevich P.V., Udod E.V. Ways to improve the security of services using JWT tokens // Economics. Information technologies. – 2023. – Vol. 50, No. 1. – P. 144-151. DOI: 10.52575/2687-0932-2023-50-1-144-151. EDN: AGOVQI.

4. Efremov M.A., Kalutskiy I.V., Tanygin M.O., Rudak I.I. The security of personal data, social networks and advertising in the global network Internet // Proceedings of the Southwest State University. Series: Control, Computer Engineering, Information Science. Medical Instruments Engineering. – 2017. – Vol. 7, No. 1 (22). – P. 27-33. – EDN: YKKKLJ

5. Keymetinova V.P., Gerasimova L.V. The collection and recycling of batteries in the city of Yakutsk // Bulatovskie chteniya. – 2020. – Vol. 6. – P. 130-132. – EDN CQTPEV.

6. Klimov K.O., Piskun G.A., Likhachevsky D.V., Alekseev V.F., Shatalova V.V. IoT data analytics using AWS IoT Analytics for environmental gas analysis // Big Data and Advanced Analytics. – 2021. – No. 7-1. – P. 131-137. EDN: SFAVIY.

7. Knaub V.V., Kovalenko A.E. The Impact of Batteries on the Environment and Human Health // Molodoy uchenyy. – 2021. – No. 3 (345). – P. 343-345.

8. Moskovkin V.M. Systems for collection and recycling of batteries and accumulators according to the European model // Tverdye bytovye otkhody. – 2022. – No. 9 (195). – P. 60-64. – EDN: MITNPL.

9. Tukaev I.R., Bespoyasova V.A. Comparison of API technologies: REST, SOAP, WebSocket, GraphQL – the choice for integration with corporate systems // Systems Engineering and Information Technologies. – 2024. – Vol. 6, No. 2 (17). – P. 78-85. DOI: 10.54708/2658-5014-SIIT-2024-no2-p78. EDN: KCYBPB.

10. Lopatko N., Kenikh A., Yakovlev A., Tian E., Pirogova V.Yu. Battery recycling in Russia and the world // Vestnik Kamchatskogo politekhnicheskogo tekhnika. – 2015. – No. 13. – P. 46-51. – EDN VYUBPN.

11. Tsibizova T.Yu., Karpunin A.A. Method of hierarchies' analysis for quality assessment of managerial processes // Modern Problems of Science and Education. – 2015. – No. 2-1. – P. 200. – EDN UHWZPP.

12. Al Rasyid M.U.H., Mubarrak M.H., Nur Hasim Ja.A. Implementation of environmental monitoring based on KAA IOT platform // Bulletin of Electrical Engineering and Informatics. – 2020. – Vol. 9, No. 6. – P. 2578-2587. DOI: 10.11591/eei.v9i6.2578. EDN: PKGAJF.

13. Hamari J., Koivisto J., Sarsa H. Does Gamification Work? – A Literature Review of Empirical Studies on Gamification // 2014 47th Hawaii International Conference on System Sciences. – Waikoloa, HI, 2014. – C. 3025-3034. DOI: 10.1109/HICSS.2014.377.

14. Jaafar H., Aminuddin N.S. Comparison between NodeMCU ESP8266 and Uno R3 for software development using Ubidots // Malaysian Journal of Bioengineering and Technology (MJBET). – 2024. – Т. 1, № 1. – С. 36-40. DOI: 10.70464/mjbet.v1i1.1320. EDN: NAJOAV.

15. Looyestyn J., Kernot J., Boshoff K., Ryan J., Edney S., Maher C. Does gamification increase engagement with online programs? A systematic review // PLoS One. – 2017. – Vol. 12, № 3. – P. e0173403. DOI: 10.1371/journal.pone.0173403. PMID: 28362821; PMCID: PMC5376078.
16. P. K. A sensor based IoT monitoring system for electrical devices using Blynk framework // Journal of Electronics and Informatics. – 2020. – Vol. 2, Iss. 3. – P. 182-187. DOI: 10.36548/jei.2020.3.005. EDN: KYTMOT.
17. Priyawati D., Rokhmah S., Utomo I.C. Website vulnerability testing and analysis of website application using OWASP // International Journal of Computer and Information System (IJCIS). – 2022. – Т. 3, № 3. – С. 142-147. DOI: 10.29040/ijcis.v3i3.90. EDN: FGAEWY.
18. Shvaika D.I., Shvaika A.I., Artemchuk V.O. Advancing IoT interoperability: dynamic data serialization using ThingsBoard // Journal of Edge Computing. – 2024. – DOI: 10.55056/jec.745. – EDN: ADATOL.
19. Van Schaik P., Ling J. Design parameters of rating scales for web sites // ACM Transactions on Computer-Human Interaction. – 2007. – Vol. 14, No. 1. – P. 4-es. DOI: 10.1145/1229855.1229859.
20. Widodo, Ali Nur Rifky. IOT-based smart trash monitoring using blynk application // Best: Journal of Applied Electrical, Science, & Technology. – 2021. – Vol. 3, No. 2. – P. 25-29. DOI: 10.36456/best.vol3.no2.4266. EDN: CLQNMR.

Рыбанов Александр Александрович, кандидат технических наук, доцент, заведующий кафедрой «Информатика и технология программирования», Волжский политехнический институт (филиал) ФГБОУ ВО Волгоградский государственный технический университет, г. Волжский, Россия

Кузминский Максим Сергеевич, студент кафедры «Информатика и технология программирования», Волжский политехнический институт (филиал) ФГБОУ ВО Волгоградский государственный технический университет, г. Волжский, Россия

Rybanov Alexander Aleksandrovich, Candidate of Technical Sciences, Associate Professor, Head of the Department of Informatics and Programming Technology, Volzhsky Polytechnic Institute (branch) of Volgograd State Technical University, Volzhsky, Russia

Kuzminsky Maxim Sergeevich, Student of the Department of Informatics and Programming, Volzhsky Polytechnic Institute (branch) of Volgograd State Technical University, Volzhsky, Russia