

ВЕБ-ПРИЛОЖЕНИЕ ПО УПРАВЛЕНИЮ БЕЗОПАСНОСТЬЮ ТРУДОВОЙ ДЕЯТЕЛЬНОСТИ НА ПРЕДПРИЯТИЯХ ПРОМЫШЛЕННОСТИ

Куканьков Е.Д.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

Научный руководитель: Телеш И.А. – к.г.н, доцент, доцент кафедры ИПиЭ

Аннотация. Разработано веб-приложение для автоматизации процессов управления безопасностью трудовой деятельности на промышленных предприятиях. Веб-приложение реализовано с использованием клиент-серверной архитектуры. Серверная часть разработана на языке Python с применением фреймворка FastAPI, клиентская часть реализована на JavaScript, а для хранения данных используется система управления базами данных PostgreSQL.

Ключевые слова: веб-приложение, промышленная безопасность, управление персоналом, FastAPI, Python, PostgreSQL, JavaScript.

Введение. Обеспечение безопасности труда является ключевой задачей для предотвращения производственного травматизма, профессиональных заболеваний и включает контроль соблюдения требований и правил техники безопасности, обучение персонала, снижение рисков и внедрение превентивных мер на промышленных предприятиях.

На рабочих местах мониторинг безопасности на предприятии осуществляется посредством использования разрозненных документов или локальных программных решений, что в свою очередь усложняет контроль выполнения корректирующих мероприятий о состоянии безопасности на предприятии.

В связи с этим одним из актуальных решений является разработка информационной системы для автоматизации учета сотрудников, регистрации нарушений техники безопасности и анализа статистических данных, связанных с производственным травматизмом на предприятии.

Основная часть. Разработанное веб-приложение построено на основе клиент-серверной архитектуры, при которой клиентская и серверная части функционируют независимо друг от друга и взаимодействуют посредством сетевых запросов. Такой подход позволяет обеспечить гибкость разработки, масштабируемость системы и удобство дальнейшей модификации функционала. Кроме того, разделение клиентской и серверной логики способствует повышению безопасности системы и облегчает процесс сопровождения программного обеспечения в процессе эксплуатации.

Дополнительным преимуществом данной архитектуры является возможность независимого обновления компонентов системы. Это означает, что изменения в серверной части могут быть реализованы без необходимости модификации клиентского приложения, что существенно упрощает сопровождение и развитие программного продукта.

Клиент-серверная архитектура является одной из наиболее распространенных моделей построения современных веб-приложений. В рамках данной архитектуры клиентская часть отвечает за отображение интерфейса пользователя и обработку пользовательских действий, тогда как серверная часть выполняет обработку запросов, реализацию бизнес-логики и взаимодействие с базой данных. Такое разделение функций позволяет повысить надежность работы системы и облегчает процесс масштабирования при увеличении количества пользователей.

Клиентская часть веб-приложения реализована с использованием языка *JavaScript* и предназначена для взаимодействия пользователя с системой через веб-интерфейс. Пользователь может работать с веб-приложением через стандартный веб-браузер без необходимости установки дополнительного программного обеспечения. Интерфейс

предоставляет средства для просмотра информации, добавления новых записей и редактирования существующих данных. Язык *JavaScript* широко применяется при разработке веб-интерфейсов и обеспечивает интерактивность и динамическое обновление содержимого страниц [1].

Серверная часть веб-приложения разработана на языке программирования *Python* с использованием веб-фреймворка *FastAPI*, предназначенного для создания высокопроизводительных *API*-сервисов [2]. Данный фреймворк позволяет эффективно обрабатывать *HTTP*-запросы и упрощает разработку серверной логики веб-приложения.

Взаимодействие между клиентом и сервером осуществляется с использованием формата *JSON*, который широко применяется при разработке веб-сервисов благодаря своей простоте и удобству обработки.

В качестве системы управления базами данных используется *PostgreSQL*, которая обеспечивает надежное хранение информации и поддержку сложных операций обработки данных [3]. В базе данных хранится информация о сотрудниках предприятия, зарегистрированных нарушениях, производственных инцидентах и принятых корректирующих мерах. Структура базы данных спроектирована таким образом, чтобы обеспечить целостность и согласованность хранимой информации. Использование реляционной модели данных позволяет эффективно выполнять выборки, формировать отчеты и анализировать статистические показатели, связанные с состоянием охраны труда.

На рисунке 1 представлена структурная схема веб-приложения по управлению безопасностью трудовой деятельности на предприятии.

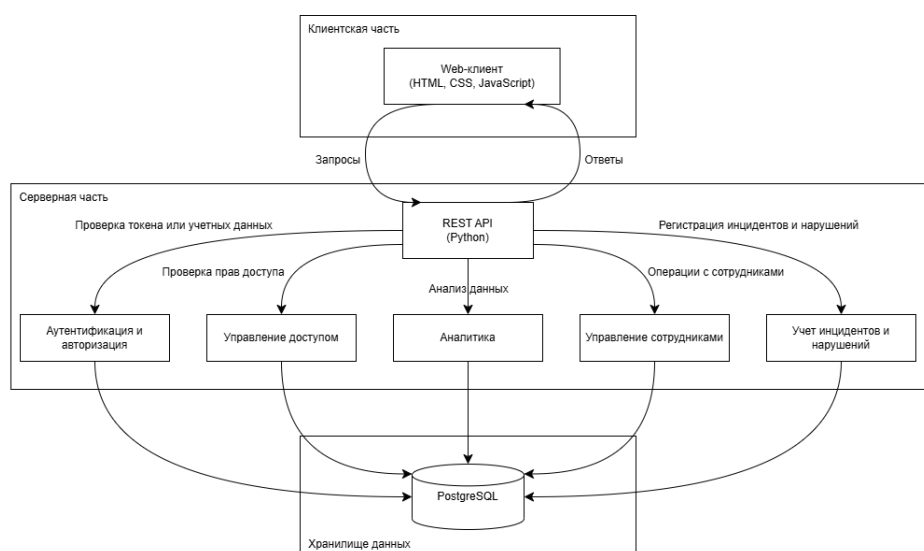


Рисунок 1 – Структурная схема веб-приложения по управлению безопасностью трудовой деятельности на предприятии

Функциональные возможности разработанного веб-приложения включают несколько основных модулей. Первый модуль разработанного веб-приложения предназначен для работы с информацией о сотрудниках. Система позволяет просматривать список работников предприятия, получать подробную информацию о каждом сотруднике, а также редактировать или добавлять новые записи. Для каждого сотрудника может храниться информация о зафиксированных нарушениях техники безопасности.

Второй модуль разработанного веб-приложения связан с учетом нарушений и производственным травматизмом. Пользователь системы может регистрировать новые случаи нарушения правил безопасности, редактировать существующие записи и просматривать историю событий. Наличие централизованной базы данных позволяет формировать полную картину происходящих на предприятии событий и анализировать причины возникновения травматизма. Это дает возможность выявлять наиболее распространенные типы нарушений и определять подразделения предприятия, где уровень риска наиболее высок. На основе

полученных данных руководство предприятия может разрабатывать дополнительные профилактические мероприятия, направленные на повышение уровня безопасности труда.

Отдельный раздел разработанного веб-приложения посвящен корректирующим мерам, которые применяются в ответ на выявленные нарушения техники безопасности. В рамках данного функционала можно фиксировать меры дисциплинарного или организационного характера, направленные на предотвращение повторения подобных ситуаций. Кроме того, система позволяет автоматически формировать документы, например приказ о выдаче нарушения, который может быть подготовлен для последующей печати.

Еще одной важной частью веб-приложения является информационная панель, на которой отображаются основные статистические показатели. На дашборде могут отображаться данные о количестве нарушений, распределении инцидентов по категориям и другие показатели, позволяющие оценить текущее состояние безопасности труда на предприятии. Визуализация информации облегчает анализ данных и помогает быстрому принятию управленческих решений. Использование графиков, диаграмм и сводных показателей позволяет пользователям системы быстро оценивать текущую ситуацию и отслеживать динамику изменений показателей безопасности труда.

Использование языка *Python* для серверной части обеспечивает гибкость разработки и широкие возможности интеграции с другими программными компонентами [4].

Заключение. Разработано веб-приложение для управления безопасностью трудовой деятельности на промышленных предприятиях. Система позволяет вести учет сотрудников, фиксировать нарушения техники безопасности, регистрировать случаи производственного травматизма и контролировать выполнение корректирующих мероприятий.

Серверная часть веб-приложения реализована на языке программирования *Python* с использованием веб-фреймворка *FastAPI*, который обеспечивает высокую производительность и удобство разработки *API*-сервисов. Клиентская часть разработана на языке *JavaScript* и обеспечивает интерактивное взаимодействие пользователя с системой через веб-интерфейс. Для хранения и обработки данных используется система управления базами данных *PostgreSQL*, обеспечивающая надежность хранения информации и поддержку сложных запросов.

Список литературы

1. *JavaScript Guide* [Электронный ресурс]. Режим доступа: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>. Дата доступа: 06.03.2026.
2. *FastAPI Documentation* [Электронный ресурс]. Режим доступа: <https://fastapi.tiangolo.com/>. Дата доступа: 05.03.2026.
3. *PostgreSQL Documentation* [Электронный ресурс]. Режим доступа: <https://www.postgresql.org/docs/>. Дата доступа: 06.03.2026.
4. *Python Documentation* [Электронный ресурс]. Режим доступа: <https://docs.python.org/>. Дата доступа: 05.03.2026.

WEB APPLICATION FOR OCCUPATIONAL SAFETY MANAGEMENT AT INDUSTRIAL ENTERPRISES

Kukankov E.D.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Telesh I.A. – Cand. of Sci., Associate Professor of the Department of Engineering Psychology and Ergonomics

Annotation. A web application was developed to automate occupational safety management processes at industrial enterprises. The web application utilizes a client-server architecture. The server-side component is developed in Python using the FastAPI framework, the client-side component is implemented in JavaScript, and the PostgreSQL database management system is used for data storage.

Keywords: web application, industrial safety, personnel management, FastAPI, Python, PostgreSQL, JavaScript.