

УДК 004.4'236

https://doi.org/10.33619/2414-2948/127/26

СИСТЕМА PUSH-УВЕДОМЛЕНИЙ В ПРОГРЕССИВНОМ ВЕБ-ПРИЛОЖЕНИИ ДЛЯ ТРАНСПОРТНОЙ КОМПАНИИ

©*Мамажакыпова Г. Т.*, ORCID: 0000-0001-5525-6837, SPIN-код: 4688-8307,

Ошский технологический университет им. М. М. Адышева,

г. Ош, Кыргызстан, gulzara.mamazhakypova@mail.ru

©*Эргешалиев Д. И.*, *Ошский технологический университет им. М. М. Адышева,*

г. Ош, Кыргызстан

A PUSH NOTIFICATION SYSTEM IN A PROGRESSIVE WEB APPLICATION FOR A TRANSPORT COMPANY

©*Mamazhakypova G.*, ORCID: 0000-0001-5525-6837, SPIN-code: 4688-8307, *Osh Technological*

University named after M. Adyshev, Osh, Kyrgyzstan, gulzara.mamazhakypova@mail.ru

©*Ergeshaliev D.*, *Osh Technological University named after M. Adyshev, Osh, Kyrgyzstan*

Аннотация. Рассматриваются принципы разработки и внедрения системы push-уведомлений в прогрессивном веб-приложении (PWA), созданном для транспортной компании “NUR Express” с использованием современных технологий: React на стороне клиента, Node.js для серверной логики и PostgreSQL для хранения данных. Описывается архитектура системы, механизмы взаимодействия компонентов и особенности реализации push-уведомлений с использованием Service Worker и Web Push API. Приводятся примеры практического применения в транспортно-логистической системе. Результаты исследования показывают, что внедрение push-уведомлений значительно повышает скорость информирования пользователей и эффективность взаимодействия с системой.

Abstract. This article examines the principles of developing and implementing a push notification system in a progressive web application (PWA) created for the NUR Express transport company using modern technologies: React on the client side, Node.js for server-side logic, and PostgreSQL for data storage. It describes the system architecture, the mechanisms for component interaction, and the specifics of implementing push notifications using Service Worker and the Web Push API. Examples of practical application in a transportation and logistics system are provided. The results of the study demonstrate that implementing push notifications significantly improves the speed of user notifications and the efficiency of interaction with the system.

Ключевые слова: PWA, push-уведомления, Service Worker, Web Push API, веб-приложение.

Keywords: PWA, push notifications, Service Worker, Web Push API, web application.

Современные веб-приложения стремятся обеспечить взаимодействие с пользователем на уровне, сопоставимом с нативными мобильными приложениями [1].

Одним из ключевых элементов этого взаимодействия являются push-уведомления, позволяющие быстро доставлять информацию без активного участия пользователя. Эта технология особенно актуальна в транспортных и логистических системах, где своевременное оповещение о статусе доставки, изменениях маршрута или задержках играет важную роль. Поэтому существует необходимость в разработке эффективной системы push-уведомлений для прогрессивных веб-приложений [2].

Актуальность данной темы определяется следующими факторами: рост использования прогрессивных веб-приложений (PWA); потребность в быстром обмене информацией в режиме реального времени; рост объёма логистических операций; необходимость снижения нагрузки на сервер за счёт асинхронных уведомлений [4].

Использование push-технологий может значительно улучшить качество обслуживания пользователей и оптимизировать бизнес-процессы.

Цель нашего исследования - является разработка и экспериментальная проверка системы push-уведомлений для прогрессивного веб-приложения транспортной компании.

Для достижения поставленной цели в рамках исследования были решены следующие задачи: проведён анализ принципов функционирования push-уведомлений в прогрессивных веб-приложениях, выполнено проектирование архитектуры системы push-уведомлений, реализован механизм подписки пользователей на уведомления, разработан и внедрён серверный механизм отправки push-уведомлений, а также проведена оценка эффективности разработанной системы.

Методы и методология исследования

При разработке системы push-уведомлений для ранее созданного прогрессивного веб-приложения (PWA) транспортной компании “NUR Express” был использован комплексный подход, сочетающий теоретические и практические методы исследования, направленные на обеспечение эффективности, масштабируемости и надежности разработанной системы (<https://707.su/CFiH>) [5].

Методологической основой для веб-приложения послужило использование технологий Service Worker и Web Push API, которые обеспечивают реализацию механизма push-уведомлений и возможность работы приложения в автономном режиме [6].

Service Worker используется в качестве промежуточного слоя между клиентским приложением и сервером, перехватывает сетевые запросы, кэширует ресурсы и обрабатывает входящие push-сообщения. Это обеспечивает стабильную работу веб-приложения даже при нестабильном интернет-соединении и повышает скорость загрузки за счет локального хранения данных [7].

Web Push API используется для доставки уведомлений с сервера на клиентское устройство пользователя. Эта технология регистрирует подписку пользователя, создает уникальный push-токен, а затем отправляет уведомления через push-сервис браузера [7].

Интеграция Service Worker и Web Push API в архитектуру PWA транспортной компании позволяет реализовать систему быстрого информирования пользователей о статусе заказа, изменениях маршрута и других логистических событиях в режиме реального времени. Это повышает эффективность взаимодействия с клиентами и снижает нагрузку на сервер благодаря асинхронной модели доставки данных.

Результаты исследования и их обсуждение

Прогрессивное веб-приложение (PWA) — это современный подход к веб-разработке, сочетающий в себе преимущества веб-сайтов и нативных приложений. Использование PWA для веб-приложения транспортной компании обеспечивает высокую производительность, возможность установки на устройство пользователя без загрузки из магазинов приложений, а также поддержку работы в автономном режиме благодаря кэшированию данных и сервисным работникам. Внедрение технологии PWA предоставляет диспетчерам, водителям и клиентам удобный доступ к системе через браузер, независимо от типа устройства. Дополнительным

преимуществом является интеграция push-уведомлений, позволяющая пользователям быстро получать обновления о статусе заказов и системных событиях.

В этом веб-приложении “NUR Express” выделяет три основные категории пользователей, получающих push-уведомления: диспетчеры, водители и клиенты. Диспетчеры работают через веб-интерфейс на персональных компьютерах, используя современные браузеры. В их задачи входит создание и обработка заказов, назначение водителей, а также мониторинг и координация транспортных операций в режиме реального времени. Водители взаимодействуют с системой через мобильные устройства, используя веб-браузеры на платформах Android и iOS. Они получают push-уведомления, содержащие информацию о новых заказах, изменениях маршрута и оповещениях о статусе их груза и транспортных средств. Клиенты используют веб-приложение для отслеживания статуса своего груза и получения уведомлений о доставке. Доступ к системе возможен как с мобильных устройств, так и с персональных компьютеров. Все категории пользователей работают в рамках единой информационной системы, но в зависимости от их роли реализован дифференцированный пользовательский интерфейс и механизм доставки уведомлений. Такой подход обеспечивает персонализированное взаимодействие и повышает эффективность управления логистическими процессами в транспортной компании.

Архитектура приложения часто определяется текущей структурой push-уведомлений браузера. В дополнение к основному веб-приложению в браузере пользователя регистрируется отдельная программа — Service Worker. Service Worker — это фоновый скрипт браузера, обеспечивающий обработку push-уведомлений, кэширование данных и выполнение сетевых запросов независимо от состояния веб-приложения. Это работает даже когда сам браузер закрыт, то есть уведомления можно получать, даже когда веб-приложение не запущено на устройстве пользователя. После регистрации в системе Service Worker подписывается на браузер пользователя для получения уведомлений, а затем на сервер отправляется специальный объект подписки. Объект подписки содержит всю необходимую информацию для отправки уведомления конкретному пользователю и состоит из уникального URL-адреса (endpoint), используемого для отправки сообщения конкретному пользователю, а также открытого ключа шифрования. Открытый ключ шифрования — это один из двух ключей шифрования на стороне сервера, предоставляемых веб-приложению [7].

Объекты подписки, получаемые сервером от клиента, сохраняются в базе данных для последующего использования. При необходимости отправки уведомления сервер извлекает соответствующую подписку и направляет запрос в push-сервис, используя уникальный endpoint-адрес. Для обеспечения безопасности применяется протокол Voluntary Application Server Identification for Web Push (VAPID), позволяющий аутентифицировать сервер посредством цифровой подписи с использованием приватного ключа. Push-сервис проверяет подлинность сервера с помощью соответствующего публичного ключа и, в случае успешной проверки, доставляет сообщение пользователю. После получения уведомления активируется Service Worker, который обрабатывает его и отображает пользователю. Таким образом, архитектура всего веб-приложения соответствует модели Клиент-Сервер. Для реализации серверной части системы управления push-уведомлениями был выбран язык JavaScript с использованием платформы Node.js. Node.js представляет собой асинхронную событийно-ориентированную среду выполнения, основанную на движке V8, обеспечивающем преобразование JavaScript в машинный код [5].

Благодаря использованию цикла событий и механизма callback-функций, Node.js позволяет эффективно обрабатывать асинхронные операции без блокировки основного потока

выполнения. Это повышает производительность серверной части и упрощает разработку масштабируемых веб-приложений [6].

Модуль web-push удобен для отправки push-уведомлений. Этот модуль реализует протокол Web Push, который описывает взаимодействие между сервером пользователя и службой push-уведомлений [7].

Кроме того, модуль web-push автоматически шифрует данные в соответствии со спецификацией шифрования сообщений для Web Push, предоставляет удобный API для отправки сообщений и включает консольную утилиту для генерации ключей шифрования, о которых говорилось ранее [8].

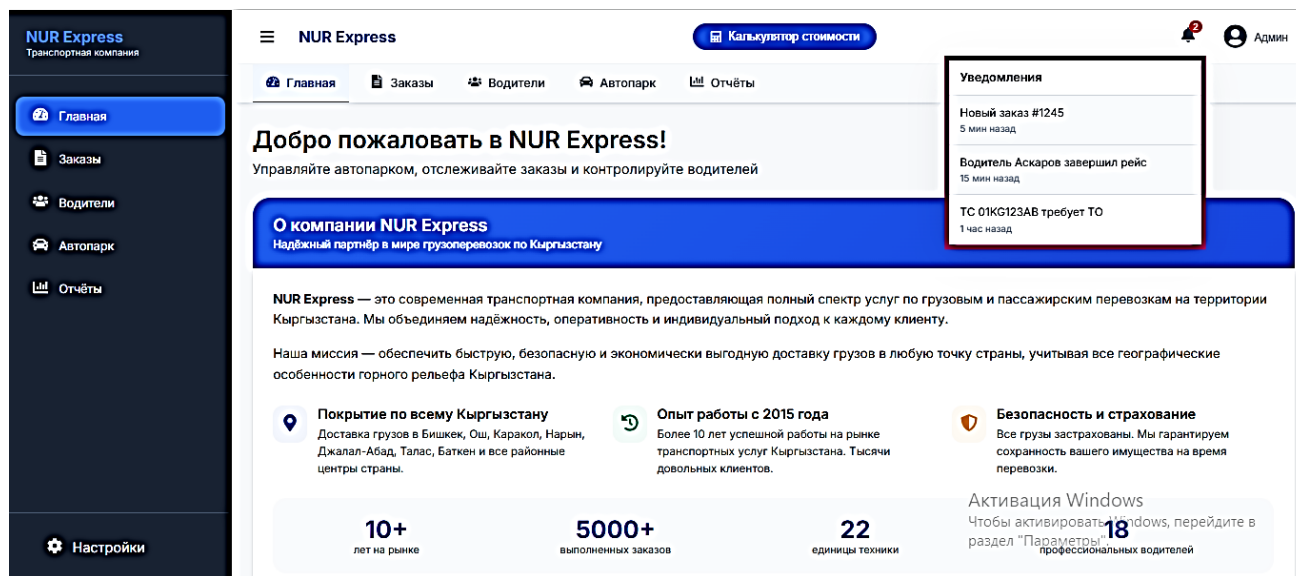


Рисунок 1. Интерфейс отображения push-уведомлений в веб-приложении

Предлагаемый веб-интерфейс приложения реализует систему push-уведомлений, отображаемую в правом верхнем углу пользовательского интерфейса. Уведомления генерируются автоматически при возникновении определенных системных событий, таких как создание нового заказа, завершение поездки водителя или необходимость технического обслуживания транспортного средства. Каждое уведомление включает краткое описание события (например, «Новый заказ №1245» или «Поездка водителя завершена») и метку времени, указывающую на время события. Это позволяет пользователю быстро отслеживать изменения в системе без необходимости обновления страницы. Помимо системы push-уведомлений, реализует механизм управления состоянием, позволяющий отслеживать текущий статус заказов, водителей и других системных объектов. Статус отражает этап бизнес-процесса и изменяется в зависимости от действий пользователя или системы. Например, можно использовать статусы заказов, такие как «создан», «получен», «обработан», «в процессе» и «завершен». При изменении статуса система автоматически записывает его в базу данных и может инициировать push-уведомление пользователю. Статусы и уведомления работают вместе для обновления информации в режиме реального времени. Статус заказа - ключевой элемент системы управления заказами в веб-приложении, обеспечивающий управление этапами выполнения, автоматизацию процессов и информирование пользователей о текущем состоянии заказа.

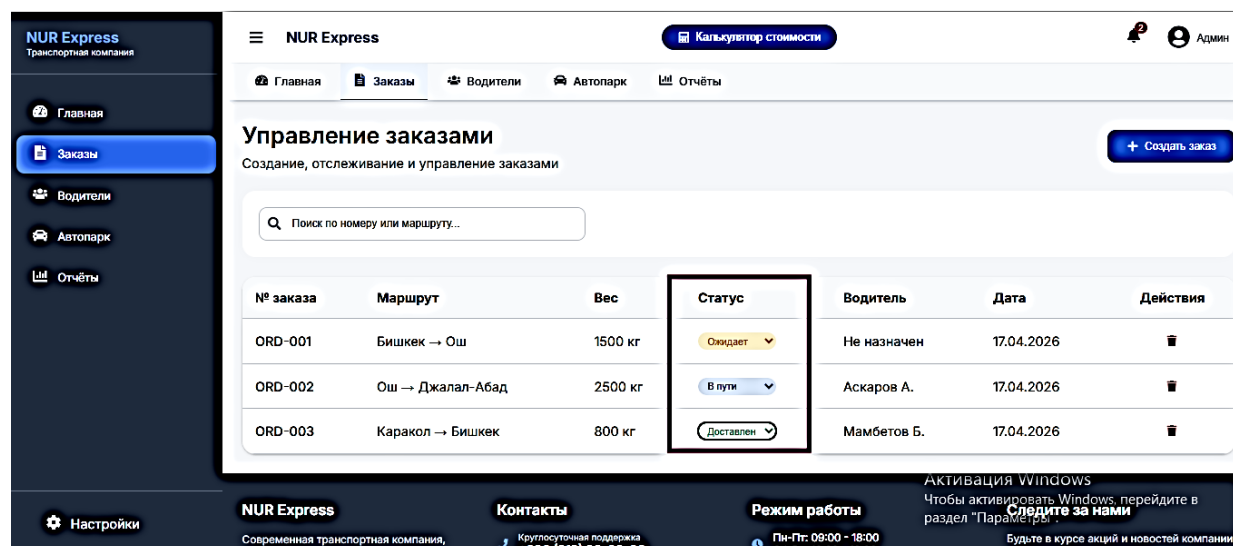


Рисунок 2. Интерфейс управления заказами в веб-приложении

Вывод

В рамках проекта была разработана система push-уведомлений, адаптированная для использования в веб-приложении транспортной компании. Основная функциональность системы направлена на обеспечение быстрого обмена информацией между участниками логистического процесса, включая диспетчеров, водителей и клиентов. Внедрение push-уведомлений позволяет автоматизировать основные этапы транспортных операций, такие как назначение заказа, уведомление об изменении статуса доставки и оповещения о чрезвычайных ситуациях. Это значительно сокращает время на предоставление информации, уменьшает необходимость ручного контроля и повышает эффективность управления транспортом. Использование механизма подписки на уровне устройства упрощает взаимодействие пользователя с системой и исключает необходимость обязательной регистрации для получения уведомлений, что особенно важно для клиентов. При этом централизованное управление уведомлениями диспетчером обеспечивает контроль и безопасность системы. В дальнейшем развитие системы может включать реализацию расширенных функций на стороне сервера, механизмов анализа эффективности уведомлений и внедрение очередей уведомлений для обеспечения стабильной работы при высокой нагрузке. Это повышает масштабируемость системы и адаптирует её к потребностям крупной транспортной компании. Таким образом, использование push-уведомлений в веб-приложении транспортной компании помогает повысить эффективность, надёжность и качество обслуживания, а также оптимизировать логистические процессы в целом.

Список литературы:

1. Поросенкова В. В., Малахов С. В., Якупов Д. О. Разработка веб-приложения для оптимизации деятельности логистики компании // Теория и практика современной науки. 2023. №5 (95). С. 163-167.
2. Курипта О. В., Наливайко И. А., Лынов Я. В. Разработка веб-ресурса. Воронеж, 2018. 101 с.
3. Мамажакыпова Г. Т., Сапарбай К. Г. Разработка веб-сайта магазина по продаже электронной техники с системой управления контентом CMS // Известия Ошского технологического университета. 2019. №1. С. 60-64.

4. Киргизова Е. В., Рубцов А. В. Web-технологии: от теории к практике. Красноярск: Сибирский федеральный университет, 2018. 160 с.
5. Заяц А. М., Васильев Н. П. Проектирование и разработка Web-приложений. Введение в Frontend и Backend разработку на JavaScript и Node.js. СПб: Лань, 2021. 120 с.
6. Мамажакыпова Г. Т., Атаханов С. Р. Разработка информационной системы регистрации пациентов для регионального медицинского учреждения // Бюллетень науки и практики. 2024. Т. 10. №4. С. 421-425. <https://doi.org/10.33619/2414-2948/101/47>
7. Федченко И. О., Намиот Д. Е. Модель информационной системы на базе push-уведомлений // International Journal of Open Information Technologies. 2015. Т. 3. №8. С. 17-29.

References:

1. Porosenkova, V. V., Malaxov, S. V., & Yakupov, D. O. (2023). Razrabotka veb-prilozheniya dlya optimizatsii deyatel'nosti logistiki kompanii. *Teoriya i praktika sovremennoj nauki*, (5 (95)), 163-167. (in Russian).
2. Kuripta, O. V., Nalivajko, I. A., & Ly'nov, Ya. V. (2018). Razrabotka veb-resursa. Voronezh. (in Russian).
3. Mamazhaky'pova, G. T., & Saporbaj, K. (2019). Razrabotka veb-sajta magazina po prodazhe e'lektronnoj tekhniki s sistemoy upravleniya kontentom CMS. *Izvestiya Oshskogo texnologicheskogo universiteta*, (1), 60-64. (in Russian).
4. Kirgizova, E. V., & Rubczov, A. V. (2018). Web-texnologii: ot teorii k praktike. Krasnoyarsk. (in Russian).
5. Zayacz, A. M., & Vasil'ev, N. P. (2021). Proektirovanie i razrabotka Web-prilozhenij. Vvedenie v Frontend i Backend razrabotku na JavaScript i Node.js. St. Petersburg. (in Russian).
6. Mamazhakupova, G., & Atakhanov, S. (2024). Development of a Patient Registration Information System for a Regional Medical Center. *Bulletin of Science and Practice*, 10(4), 421-425. (in Russian). <https://doi.org/10.33619/2414-2948/101/47>
7. Fedchenko, I. O., & Namiot, D. E. (2015). Model' informacionnoj sistemy' na baze push-uedomlenij. *International Journal of Open Information Technologies*, 3(8), 17-29. (in Russian).

Поступила в редакцию
17.04.2026 г.

Принята к публикации
28.04.2026 г.

Ссылка для цитирования:

Мамажакыпова Г. Т., Эргешалиев Д. И. Система push-уведомлений в прогрессивном веб-приложении для транспортной компании // Бюллетень науки и практики. 2026. Т. 12. №6. С. 213-218. <https://doi.org/10.33619/2414-2948/127/26>

Cite as (APA):

Mamazhakupova, G., & Ergeshaliev, D. (2026). A Push Notification System in a Progressive Web Application for a Transport Company. *Bulletin of Science and Practice*, 12(6), 213-218. (in Russian). <https://doi.org/10.33619/2414-2948/127/26>