

Центральным элементом системы является модуль управления перевозками. На соответствующей странице отображаются все текущие перевозки с детальной информацией по каждой: статус выполнения, адреса отправления и назначения, перевозимые товары и планируемое время доставки. Система поддерживает несколько статусов перевозок, включая «создан», «в пути», «завершен» и «ошибка».

Процесс создания новой перевозки представляет собой многоэтапную процедуру. Пользователь выбирает города отправления и назначения (из списка популярных маршрутов или вручную), указывает дату и время через всплывающий календарь, выбирает товары из каталога с указанием количества, и назначает подходящего водителя из числа доступных. Все этапы оформления интуитивно понятны и сопровождаются визуальными подсказками.

Завершает функционал системы раздел «Товары», который представляет собой каталог всей продукции, доступной для перевозки. Для каждого товара указывается наименование, подробное описание, габаритные размеры в сантиметрах, текущее количество на складе и вес в килограммах. Этот модуль обеспечивает актуальность информации о товарных остатках и упрощает процесс формирования грузов для перевозок.

Заключение. Разработка и внедрение специализированного веб-приложения для автоматизации работы логистического отдела является стратегически важным шагом для современных предприятий, стремящихся к повышению операционной эффективности, снижению издержек и усилению контроля над цепью поставок. В рамках данного проекта было успешно спроектировано, реализовано и протестировано комплексное программное решение, доказавшее свою практическую значимость и соответствие поставленным задачам.

Внедрение системы позволит сократить время на планирование доставок и повысит прозрачность логистических операций за счёт автоматизации и отслеживания статусов в реальном времени. Это приведёт к снижению операционных издержек и исключению ошибок, вызванных ручным управлением данными.

Список цитируемых источников

1. Муромцев, В. В. Информационные технологии в профессиональной деятельности : учебник и практикум / В. В. Муромцев, А. В. Муромцева. — Москва ; Вологда : Инфра-Инженерия, 2023. — 384 с.

УДК 004.42

К. Ю. Матусевич

Учреждение образования «Барановичский государственный университет», Барановичи, Республика Беларусь

Научный руководитель О. Д. Кравчук

РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ НА REACT NATIVE ДЛЯ УПРАВЛЕНИЯ ГРУЗОПЕРЕВОЗКАМИ

Введение. Современная логистика характеризуется высокой динамичностью и требовательностью к оперативности управления процессами грузоперевозок. В условиях роста конкуренции и повышения стандартов сервиса, предприятиям необходимы инструменты, обеспечивающие мгновенный обмен информацией между диспетчерскими центрами и исполнителями на линии. Одним из наиболее эффективных решений данной задачи является внедрение мобильных приложений, которые предоставляют водителям прямой доступ к заданиям, маршрутам и средствам коммуникации, а логистам — инструменты для контроля в режиме реального времени [1].

Объектом разработки является мобильное приложение для управления грузоперевозками предприятия ОАО «Торгмаш», предназначенное для оперативного взаимодействия водителей с логистическим отделом, получения и выполнения заданий по доставке, отслеживания маршрутов, контроля статусов перевозок, а также оперативной коммуникации через встроенную систему чатов.

Предметом исследования служит кроссплатформенное мобильное приложение, интегрированное с корпоративной веб-системой управления логистикой, реализованное на платформе React Native с использованием современных подходов к разработке мобильных приложений, включая работу с геолокацией, push-уведомлениями и offline-режимом.

Цель разработки: создание интуитивного и надежного мобильного приложения, которое обеспечивает водителей централизованным доступом к актуальной информации о заданиях, упрощает процесс выполнения перевозок за счет интеграции с картографическими сервисами, улучшает координацию с логистическим отделом через мгновенные уведомления и чат, а также повышает прозрачность и контролируемость процесса доставки в реальном времени.

Основная часть. Выбор средств разработки был обусловлен необходимостью создания комплексного решения, объединяющего веб- и мобильные компоненты. Для серверной части использовался микрофреймворк Flask, обеспечивающий гибкость и быстрое развертывание RESTful API, а для хранения данных — документоориентированная СУБД MongoDB, идеально подходящая для работы со сложными иерархическими структурами. Веб-интерфейс для логистов реализован на React с использованием сборщика Vite, что позволило создать производительный и удобный интерфейс.

Для реализации мобильного приложения был выбран React Native — современный фреймворк с открытым исходным кодом, разработанный и поддерживаемый компанией Meta. Он позволяет создавать кроссплатформенные мобильные приложения для iOS и Android с использованием единой кодовой базы на языке JavaScript, что значительно упрощает процесс разработки и сопровождения проекта.

React Native предоставляет разработчикам возможность создавать мобильные приложения для iOS и Android на единой кодовой базе, что значительно ускоряет процесс разработки и снижает затраты на поддержку. Благодаря встроенному механизму взаимодействия с нативными компонентами платформ, фреймворк обеспечивает высокую производительность и позволяет реализовывать сложные интерфейсы с сохранением нативного пользовательского опыта. Его активное сообщество, богатая экосистема библиотек и поддержка горячей перезагрузки делают разработку более гибкой, удобной и адаптированной к современным требованиям мобильной среды [2].

Мобильное приложение обеспечивает водителей удобным и эффективным инструментом для управления логистическими заданиями. Доступ к системе осуществляется через защищенную авторизацию с использованием уникальных учетных данных, выдаваемых при зачислении в штат компании. Для обеспечения совместимости требуется устройство с операционной системой Android версии 6.0 и выше.

После успешной аутентификации пользователь попадает на главный экран приложения — «Перевозки», который представлен на рисунке 1.

В случае отсутствия активных заданий система отображает соответствующее информационное сообщение, что исключает недопонимание и сокращает время на проверку статусов.

Каждое задание содержит детализированную информацию о перевозке: номера и количество коробок, адреса погрузки и доставки, временные рамки. Для просмотра дополнительных сведений предусмотрена интерактивная кнопка-стрелка.

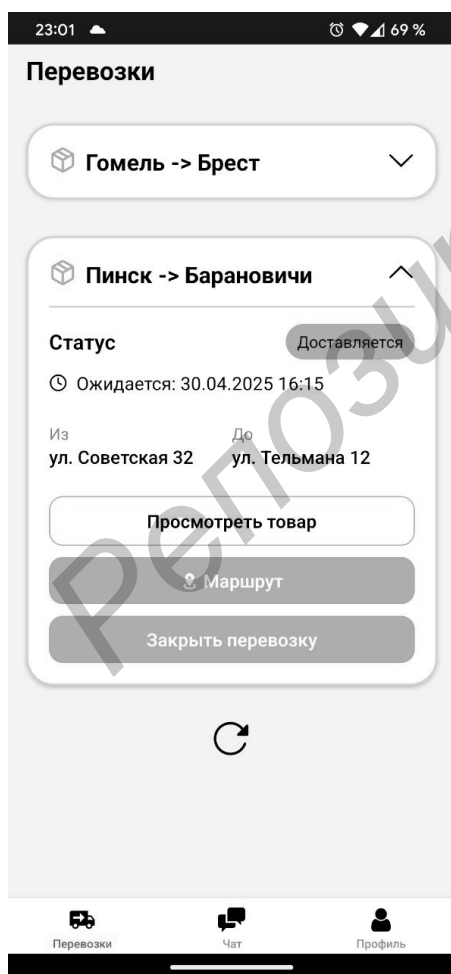


Рисунок 1 — Страница перевозок

Ключевой особенностью является интеграция с картографическими сервисами: при нажатии кнопки «Маршрут» система перенаправляет водителя в приложение Google Maps (при его наличии) или веб-версию сервиса. Построение маршрута осуществляется через API OpenStreetMap, который преобразует текстовые адреса в географические координаты и предоставляет оптимальные пути следования.

Статусы перевозок («в пути», «доставлен», «не принят» и др.) обновляются в реальном времени. Завершение задания возможно только по истечении часа от планового времени доставки, что предотвращает преждевременное закрытие заказов. При подтверждении операции перевозка автоматически архивируется в историю.

Для обеспечения полной информационной поддержки реализованы два дополнительных модуля:

1. Карточка товара — содержит краткое описание груза, его характеристики и особенности.
2. Встроенный чат — позволяет оперативно связаться с логистом для уточнения деталей. Интерфейс чата адаптирован под мобильные устройства.

Заключение. Разработанное мобильное приложение на React Native стало ключевым элементом комплексной системы автоматизации логистических процессов предприятия ОАО «Торгомаш». Реализованное решение успешно интегрировано с существующей веб-платформой и обеспечивает полный цикл управления грузоперевозками — от получения заданий до фиксации результатов доставки.

Наиболее значимыми результатами внедрения стали:

1. Сокращение времени на коммуникацию между водителями и логистами за счет встроенного чата и системы push-уведомлений.
2. Повышение точности планирования маршрутов благодаря интеграции с картографическими сервисами.

3. Увеличение прозрачности процессов доставки за счет системы статусов и автоматического архивирования выполненных перевозок.

4. Снижение количества ошибок при обработке грузов благодаря детализированной информации о товарах.

Применение технологического стека на основе React Native доказало свою эффективность для решения промышленных задач в логистике. Кроссплатформенный подход позволил обеспечить одинаковое качество работы на устройствах iOS и Android при сокращении затрат на разработку и поддержку.

Список цитируемых источников

1. *Новаков, А. А.* Логистика в деталях: учебное пособие / А. А. Новаков. — Москва ; Вологда : Инфра-Инженерия, 2021. — 528 с.
2. *Стефанов, С.* React. Быстрый старт, 2-е изд. / С. Стефанов. — СПб : Питер, 2023. — 304 с.

УДК 004.8

М. М. Мультиан

Учреждение образования «Барановичский государственный университет», Барановичи, Республика Беларусь

Научный руководитель О. Д. Кравчук

РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА ДЛЯ ВЫЯВЛЕНИЯ ТЕКСТОВЫХ ЗАИМСТВОВАНИЙ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Введение. Активное развитие цифровых образовательных ресурсов и увеличение объёмов электронных текстовых материалов обуславливают необходимость создания эффективных систем для проверки оригинальности контента. Проблема академического плагиата остается одной из наиболее актуальных в современной образовательной среде [1]. Традиционные системы антиплагиата часто ограничены поиском дословных совпадений, в то время как современные методы модификации текста, такие как замена символов, синонимизация и перефразирование, требуют применения более сложных алгоритмов анализа на основе искусственного интеллекта.

Целью исследования является разработка программного комплекса для автоматизированного выявления текстовых заимствований с применением методов искусственного интеллекта, способного эффективно обнаруживать как прямые заимствования, так и скрытые модификации текста.

Объектом исследования выступает процесс обработки, анализа и сравнения текстовых данных, который включает:

1. Векторизацию текста — преобразование семантических единиц текста в числовые векторы в многомерном пространстве с помощью нейросетевых моделей, что позволяет улавливать их смысловое содержание.
2. Семантический анализ — оценку смыслового содержания текста, выявление основной темы, тональности и ключевых концепций, что критически важно для обнаружения парафразирования.
3. Сравнение документов на основе нейросетевых моделей — вычисление метрик семантического сходства (например, косинусной близости) между векторными представлениями текстов для определения степени их смысловой схожести даже при отсутствии дословных совпадений.

Основная часть. Разрабатываемый программный комплекс представляет собой десктопное приложение, реализованное на платформе .NET с использованием языка C# и технологии Windows Forms. Архитектура системы построена по модульному принципу и включает следующие ключевые компоненты:

1. Модуль обработки входных данных обеспечивает приём и предварительную обработку текстовых документов различных форматов (DOCX, TXT). Функционал модуля включает: автоматическое определение кодировки текстовых файлов, извлечение текстового содержимого из документов MS Word с сохранением структурной целостности, предварительную очистку текста от служебных символов и элементов форматирования, нормализацию текста (приведение к нижнему регистру, удаление лишних пробелов).
2. Модуль лингвистического анализа осуществляет глубокую обработку текста с использованием алгоритмов NLP: токенизацию текста на предложения и отдельные слова, стемминг и лемматизацию для приведения слов к нормальной форме, выявление и коррекцию подменённых символов (замена букв кириллицы на латинские аналоги и наоборот), синтаксический анализ структуры предложений.
3. Модуль векторного представления текста реализует преобразование текста в числовые векторы с использованием предобученных нейросетевых моделей. Модуль обеспечивает: интеграцию с Python-скриптами для использования библиотек машинного обучения, применение модели для генерации семантических векторных представлений предложений, кэширование эмбедингов для ускорения последующих проверок.
4. Подсистема хранения данных и реализована на СУБД MySQL. Схема базы данных включает: таблицу для хранения метаданных об эталонных документах, таблицу для хранения предложений и их бинарных векторных представлений и таблицу для ведения истории проведённых проверок. Для обеспечения высокой производительности применены оптимизированные индексы.