

МАРКОВИЧ А. С., СОЧКО С. С., СЕМЯШКИНА А. В.
СИСТЕМА СОПРОВОЖДЕНИЯ ЗАПИСИ И ОФОРМЛЕНИЯ НА
ПРАКТИЧЕСКУЮ ПОДГОТОВКУ НА ВЫПУСКАЮЩЕЙ КАФЕДРЕ

УДК 004.9:378:004.65, ГРНТИ 50.49.37

Статья поступила в редакцию 18.06.2026

Система сопровождения записи
и оформления на практическую
подготовку на выпускающей кафедре

System for tracking registration and
processing arrangements for practical
training at the graduating department

**А. С. Маркович¹,
С. С. Сочко², А. В. Семяшкина²**

**A. S. Markovich¹,
S. S. Sochko², A. V. Semyashkina²**

¹Санкт-Петербургский
государственный университет
аэрокосмического приборостроения,
г. Санкт-Петербург;

²Ухтинский государственный
технический университет, г. Ухта

¹Saint Petersburg State University of
Aerospace Instrumentation (SUAI),
Saint Petersburg;

²Ukhta State Technical University,
Ukhta

В статье представлены результаты разработки веб-ориентированной информационной системы сопровождения записи и оформления на практическую подготовку на выпускающей кафедре. Рассмотрены основные проблемы ручного документооборота: избыточные временные затраты, ошибки при обработке данных и нагрузка на сотрудников кафедры. Описана трёхзвенная архитектура системы на стеке Node.js + Express / React + TypeScript / PostgreSQL, спроектирована реляционная база данных из 20 взаимосвязанных таблиц. Реализованы модули авторизации, выбора места практики, автоматического формирования дневника и приказа о допуске. Показано, что внедрение системы позволяет централизовать данные о студентах, практиках и предприятиях, снизить трудоёмкость

The article presents the results of developing a web-based information system to manage the registration and processing of practical training arrangements at a graduating department. It examines the key issues associated with manual document workflows: excessive time consumption, data processing errors, and the administrative burden placed on department staff. A three-tier system architecture based on the Node.js + Express / React + TypeScript / PostgreSQL stack is described, and a relational database comprising 20 interconnected tables has been designed. Modules for authorization, internship placement selection, and the automated generation of internship logs and authorization orders have been implemented. It is demonstrated that the system's deployment enables the centralization of data regarding students, internships, and host

оформления документов и обеспечить прозрачный контроль статусов на всех этапах практической подготовки.

organizations, reduces the administrative workload associated with documentation, and ensures transparent status tracking throughout all stages of practical training.

Ключевые слова: *практическая подготовка студентов, информационная система, автоматизация образовательного процесса, веб-приложение, PostgreSQL, React, Node.js*

Keywords: *practical training, information system, educational process automation, web application, PostgreSQL, React, Node.js*

Введение

Автоматизация административных процессов в системе высшего образования приобретает всё большее значение на фоне роста числа обучающихся и усложнения нормативных требований к организации практической подготовки. Согласно действующему Положению о практической подготовке обучающихся ФГБОУ ВО «УГТУ» [1], каждый студент за период обучения проходит не менее четырёх видов практик – учебную, две производственные и преддипломную. Это порождает значительный объём документооборота: распределение по предприятиям, оформление договоров, формирование приказов, назначение руководителей, ведение дневников.

На кафедре Вычислительной техники, информационных систем и технологий (далее – ВТИСиТ) данные процессы до недавнего времени выполнялись вручную специалистом по учебно-методической работе (далее – УМР).

Анализ предметной области выявил три системных проблемы:

1. Избыточная трудоёмкость ручного заполнения типовых документов;
2. Рассредоточенность информации по различным источникам (учебные планы, рабочие программы, стенды, электронная почта), что провоцирует повторяющиеся студенческие запросы;
3. Высокий риск ошибок при ручном переносе данных о студентах, группах и предприятиях.

Целью является разработка веб-ориентированной информационной системы сопровождения записи и оформления на практическую подготовку на выпускающей кафедре, автоматизирующей весь жизненный цикл сопровождения практики: от публикации заявок предприятий до формирования итоговых документов.

Для достижения цели решались следующие задачи:

- анализ предметной области и выявление функциональных требований;
- проектирование реляционной базы данных;

- разработка серверного и клиентского приложений;
- тестирование и оценка соответствия требованиям.

Обзор аналогов и литературы

Задача автоматизации организационных процессов в вузах относится к классу задач управления образовательными информационными системами. Анализ существующих аналогов – систем «Деканат» [7], 1С:Университет, модулей LMS Moodle, а также систем автоматизации подготовки документов учебного процесса [8] – показал, что они либо не поддерживают специфику практической подготовки (распределение по заявкам, многоролевое согласование), либо требуют значительных лицензионных затрат и адаптации под конкретный регламент кафедры.

Практика автоматизации учёта деятельности сотрудников кафедр в отечественных вузах (на примере НГИЭУ [9]) демонстрирует, что централизованное хранение данных позволяет сократить трудозатраты на подготовку отчётности и повысить точность документов. Это подтверждает целесообразность разработки собственного инструмента, тесно интегрированного с внутренними регламентами конкретного университета.

С технологической точки зрения современные веб-приложения подобного класса строятся по трёхзвенной клиент-серверной архитектуре с REST-интерфейсом. Сервис-ориентированный подход и REST API широко применяются при разработке информационных систем университетов, включая порталы расписания и управления студенческими данными [10, 14]. Компонентный подход React и статическая типизация TypeScript обеспечивают поддерживаемость и тестируемость кода [11, 12]. PostgreSQL как объектно-реляционная СУБД предоставляет развитый механизм ограничений целостности и высокую производительность на смешанных рабочих нагрузках [15].

Логическое проектирование базы данных на основе принципов нормализации [13] является ключевым этапом, определяющим качество модели данных и отсутствие аномалий при работе с системой. Нормализация до 3НФ, применённая в данной работе, позволяет исключить избыточность и обеспечить целостность данных.

Моделирование бизнес-процесса

Объектом автоматизации является процесс сопровождения практической подготовки на кафедре ВТИСиТ. Участники процесса: студент, специалист УМП, руководитель практики от кафедры, отдел практик университета. Диаграмма потоков данных (DFD) первого уровня, построенная в CASE-средстве Visual Paradigm представлена на Рисунке 1.

На основе моделирования выявлены девять ключевых процессов системы.

В Таблице 1 приведены функциональные требования к разрабатываемой информационной системе.

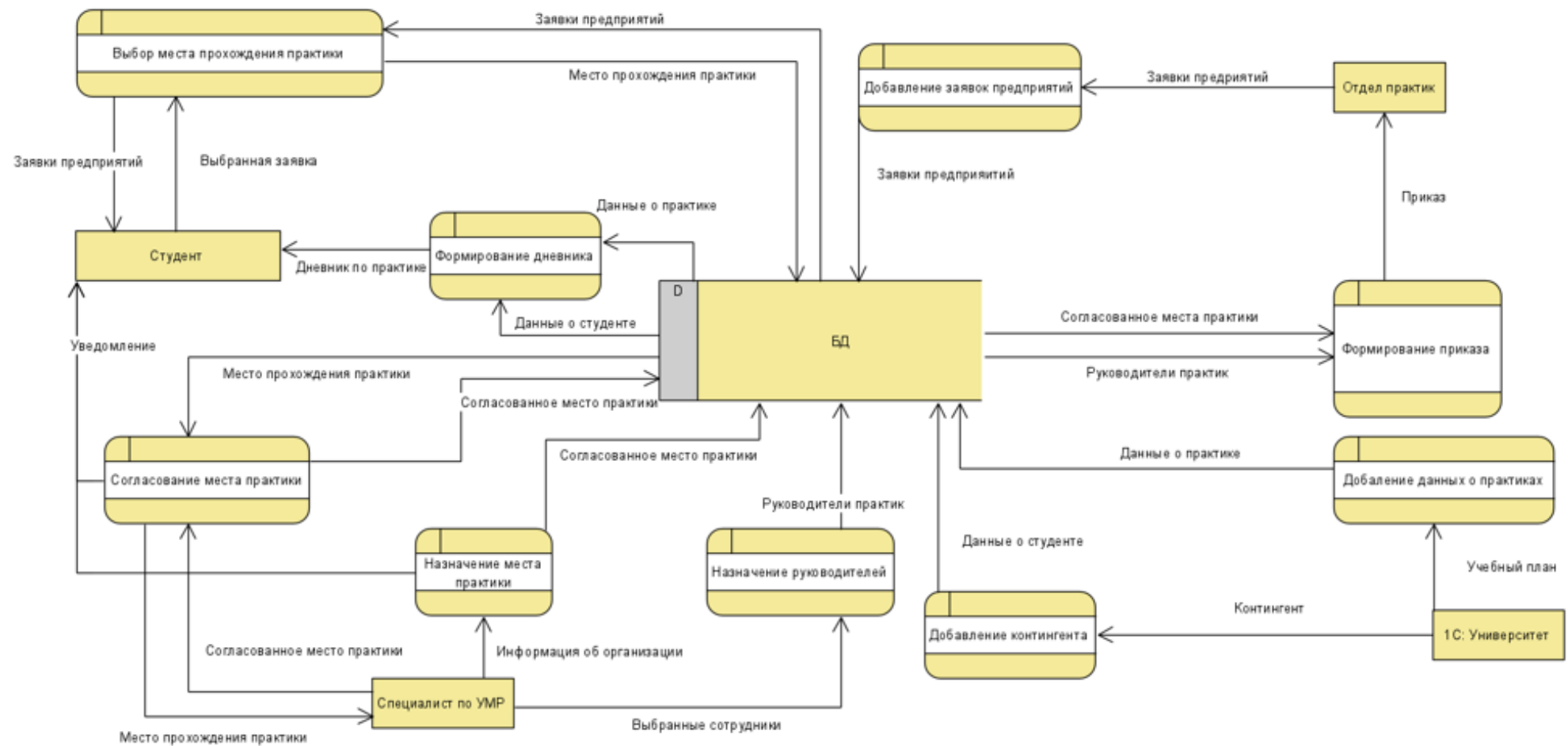


Рисунок 1. Контекстный уровень DFD-диаграммы «Как будет»

Таблица 1. Функциональные требования к ИС

№	Функциональное требование	Пользователь
1	Авторизация по номеру зачётки и паспорта	Студент
2	Просмотр информации о предстоящей практике (вид, тип, сроки, компетенции)	Студент
3	Выбор предпочитаемой заявки предприятия из опубликованного списка	Студент
4	Автоматическое формирование дневника практической подготовки (PDF)	Студент
5	Согласование / отклонение выбора студента	Специалист УМП
6	Ручное закрепление студента за заявкой при необходимости	Специалист УМП
7	Формирование приказа о допуске на практическую подготовку	Специалист УМП
8	Добавление и редактирование заявок предприятий	Отдел практик
9	Управление справочниками (кафедры, специальности, виды практик и др.)	Администратор

Архитектура системы

Разработанная система реализует трёхзвенную клиент-серверную архитектуру. Все компоненты развёрнуты на едином физическом сервере; взаимодействие фронтенда и бэкенда осуществляется исключительно через REST API посредством защищённого HTTPS-соединения. (Рисунок 2)

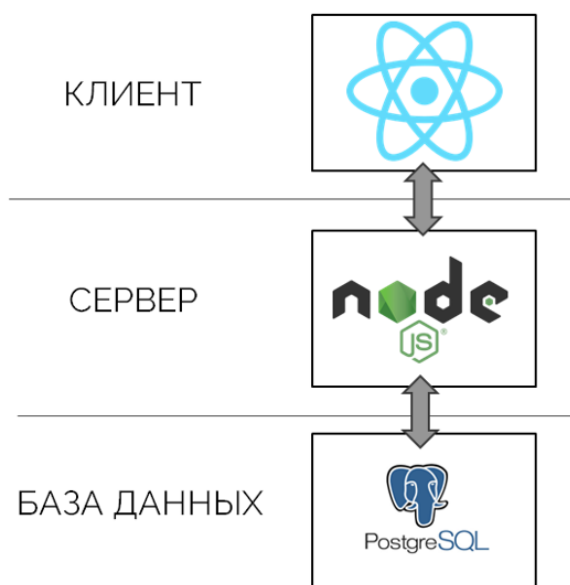


Рисунок 2. Технологический стек и архитектура системы

Модель базы данных

Реляционная база данных спроектирована в среде Visual Paradigm с использованием методологии нормализации до третьей нормальной формы [13]. База включает 20 таблиц: 9 основных сущностей предметной области и 11 справочников, промежуточных таблиц. Такой подход соответствует принципам проектирования реляционных СУБД [3] и обеспечивает отсутствие аномалий обновления и удаления данных (Рисунок 3).

Связи типа «многие ко многим» (студент – заявка, практика – компетенция, практика – руководитель) реализованы через промежуточные таблицы с дополнительными атрибутами – статусом согласования и лимитом мест. Ссылочная целостность обеспечивается каскадными внешними ключами (FOREIGN KEY ... ON DELETE RESTRICT). Для быстрого поиска по шифру студента и наименованию организации созданы индексы B-tree.

Серверная часть

Серверное приложение построено на Node.js 20 с фреймворком Express 4 [5]. API разбит на доменные роутеры: /auth, /students, /practices, /applications, /documents. Каждый роутер содержит middleware-цепочку: валидация входных данных → аутентификация JWT-токена → выполнение бизнес-логики → формирование JSON-ответа. Данный подход соответствует архитектурным принципам REST API [10], обеспечивающим гибкость и масштабируемость системы.

Аутентификация реализована по схеме «номер зачётки + номер паспорта»; после успешной проверки сервер генерирует JWT-токен с временем жизни 24 часа и полезной нагрузкой, содержащей идентификатор и роль пользователя (Рисунок 4).

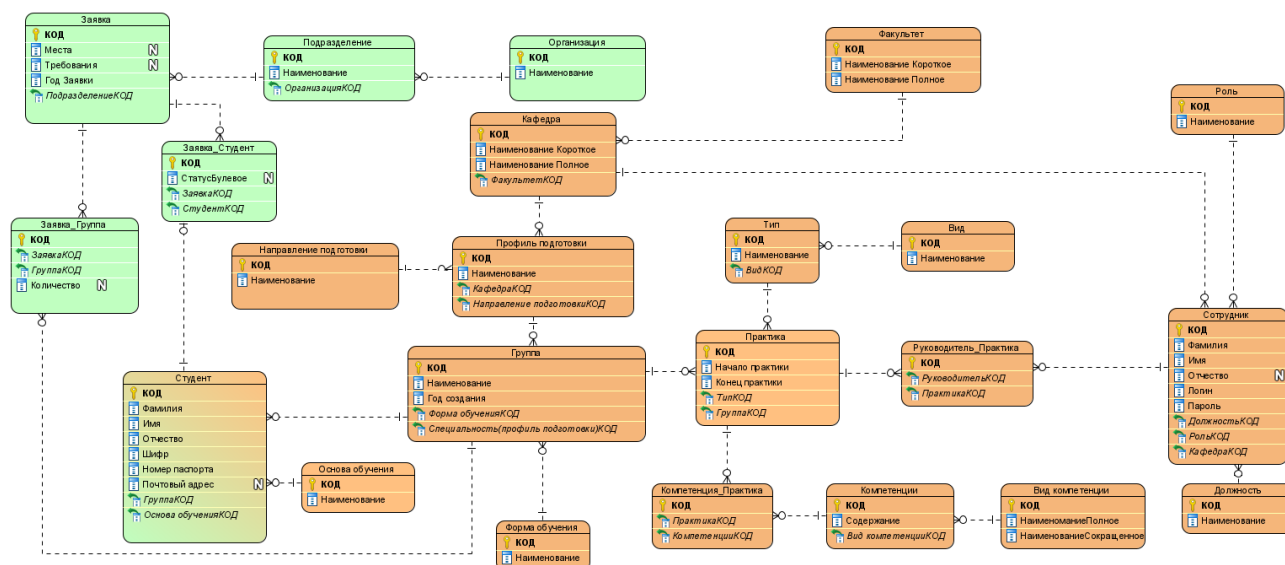


Рисунок 3. Логическая модель данных системы

Рисунок 4. Форма входа

Рисунок 5. Личный кабинет студента

Применение TypeScript [12] для строгой типизации кода обеспечивает надёжность и поддерживаемость серверного приложения.

Клиентская часть и формирование документов

Клиентская часть разработана на React 18 с TypeScript [6, 11], что обеспечивает высокопроизводительный компонентный SPA-интерфейс с виртуальным DOM. Компонентная архитектура React позволяет повторно использовать элементы интерфейса и упрощает поддержку кода.

При запросе дневника система агрегирует из базы персональные данные студента, реквизиты практики (вид, тип, сроки), перечень компетенций, сведения о руководителях и организации, и автоматически формирует документ .docx (Рисунок 6).

МИНОБРАЗОВАНИЯ РОССИИ Федеральное государственное бюджетное образовательное учреждение высшего образования «Ухтинский государственный технический университет» (УГТУ)	
ДНЕВНИК ПО ПРАКТИЧЕСКОЙ ПОДГОТОВКЕ	
Фамилия	Наркович
Имя	Андрей
Отчество	Сергеевич
Шифр	200000
Направление подготовки (специальность, профессия)	09.03.02 Информационные системы и технологии
Профиль подготовки	Информационные системы и технологии
Курс	3
Группа	ИСТ-1-230-5
Факультет (институт)	Факультет экономики, управления и информационных технологий

Компонент образовательной программы	Практика
Вид практики	Производственная практика
Тип практики	Производственная технологическая практика
ПМ	
Сроки практики	2026-06-24 - 2026-07-21
Количество часов (недель) акад.час./астроном.час.	
Место практической подготовки (наименование профильной организации или УГТУ)	
Регион, город (поселок)	

Рисунок 6. Форма для получения дневника по практике

План прохождения практики

1. Ознакомиться с информацией о практике.
2. Выбрать место прохождения практики. Перейти к заявкам
 1. Предприятие по заявкам от университета.
 2. Самостоятельный поиск предприятия. Для прохождения практической подготовки на предприятии не из списка заявок необходимо оформить договор.
[Форму для заполнения можно получить тут.](#)
 3. Прохождение практики на кафедре или в подразделениях университета.
3. Получить дневник по практической подготовке и задания от руководителя.
Получить дневник по практической подготовке
4. Сдать отчет по практике и дневник.

Рисунок 7. Сформированный дневник

По результатам тестирования время генерации дневника на сервере с 2 ядрами CPU и 4 ГБ RAM составляет менее 2 секунд. Приказ о допуске формируется специалистом УМР одним действием после согласования мест практики всех студентов группы.

Реализация и результаты

По итогам разработки и тестирования система обеспечивает выполнение всех девяти функциональных требований. Реализованы четыре пользовательские роли: студент, специалист УМР, руководитель практики, администратор. Каждой роли соответствует собственный набор экранных форм и REST-эндпоинтов.

База данных из 20 таблиц охватывает все сущности предметной области. Нормализация до 3НФ исключает аномалии обновления [13]. Архитектура

позволяет без изменения схемы добавлять новые учебные годы, специальности и организации, что обеспечивает масштабируемость на многолетнюю перспективу.

Система рассчитана на круглосуточный режим работы (365/24/7). Минимальные требования к серверу: процессор x64 с 2 ядрами, ОЗУ 4 ГБ, дисковое пространство 15 ГБ. Клиенту достаточно любого современного браузера. Тестирование при одновременной работе 30 пользователей не выявило деградации производительности: среднее время ответа REST API – менее 150 мс.

Среди перспектив развития – интеграция с корпоративными ИС университета (1С:Университет) для автоматического импорта данных о студентах и учебных планах, а также добавление модуля для руководителей практики с возможностью загрузки рабочих программ и индивидуальных заданий.

Заключение

В работе решена актуальная задача автоматизации управления практической подготовкой студентов на выпускающей кафедре университета. Разработана и внедрена веб-ориентированная ИС «Кабинет практической подготовки» на трёхзвенной архитектуре (React + TypeScript / Node.js + Express / PostgreSQL). Спроектирована нормализованная реляционная база данных из 20 таблиц, охватывающая все объекты предметной области. Реализованы модули авторизации, выбора места практики с многоролевым согласованием, автоматического формирования дневника и приказа о допуске.

Система обеспечивает централизованное хранение и оперативный доступ к данным, снижает трудоёмкость оформления документов и устраняет информационную рассредоточенность, характерную для ручного документооборота.

Список использованных источников и литературы

1. Положение о практической подготовке обучающихся: положение / ФГБОУ ВО «УГТУ». – Ухта: УГТУ, 2020. – 25 с.
2. Рочев К.В. Информационные технологии. Анализ и проектирование информационных систем: учеб. пособие. – Ухта: УГТУ, 2018. – 139 с.
3. Илюшечкин В.М. Основы использования и проектирования баз данных. – М.: Юрайт, 2013. – 224 с.
4. Рочев К. В. Классификация средств графического моделирования для разработки информационных систем // Информационные технологии в управлении и экономике. 2024. №1. Режим доступа: <http://itue.ru/Issue/Article/275>
5. Документация по Node.js [Электронный ресурс]. – URL: <https://nodejs.org/docs/latest/> (дата обращения: 10.12.2024).
6. Документация по React [Электронный ресурс]. – URL: <https://ru.legacy.reactjs.org/docs/> (дата обращения: 10.12.2024).
7. Мазарчук М. С., Тыжненко Ю. В. Автоматизированная информационная система «Деканат» // Объектные системы. 2011. № 1. С. 53-58.

URL: <https://cyberleninka.ru/article/n/avtomatizirovannaya-informatsionnaya-sistema-dekanat>

8. Матюшенко В. А., Филон В. Ю., Белодед Н. И. Информационная система автоматизации подготовки документов учебного процесса // Системный анализ и прикладная информатика 2016. № 2. С. 66-70. URL: <https://cyberleninka.ru/article/n/informatsionnaya-sistema-avtomatizatsii-podgotovki-dokumentov-uchebnogo-protssessa>

9. Астахова Т. Н., Бердникова А. А., Бенбен Д. В., Лямин А. С. Автоматизация учёта научной и учебно-методической деятельности преподавателя / International Journal of Open Information Technologies № 2. С. 117-125. URL: <https://cyberleninka.ru/article/n/avtomatizatsiya-ucheta-nauchnoy-i-uchebnometodicheskoy-deyatelnosti-prepodavatelya>

10. Файзрахманова К. Э. Сервис-ориентированная архитектура и технологическая реализация веб-сервиса // Вестник науки 2026. № 4 С. 812-818. URL: <https://cyberleninka.ru/article/n/servis-orientirovannaya-arhitektura-i-tehnologicheskaya-realizatsiya-veb-servisa>

11. Сергачева М. А., Михалевская К. А. Анализ фреймворков для разработки современных веб-приложений // Кронос: естественные и технические науки 2020 № 2. С. 35-39. URL: <https://cyberleninka.ru/article/n/analiz-freymvorkov-dlya-razrabotki-sovremennyh-veb-prilozheniy>

12. Хайруллин Д. А., Богданов М. Р. Изучение использования TypeScript в React-приложениях, преимущества статической типизации // Международный журнал гуманитарных и естественных наук 2024. № 12-3. С. 233-237. URL: <https://cyberleninka.ru/article/n/izuchenie-ispolzovaniya-typescript-v-react-prilozheniyah-preimuschestva-staticheskoy-tipizatsii-podhody-k-tipizatsii-komponentov>

13. Игнатюк В. А., Сторожок Е. А. Логическое проектирование реляционной базы данных на основе принципов нормализации // Территория новых возможностей. Вестник Владивостокского государственного университета 2011 №. 2. С. 101-105. URL: <https://cyberleninka.ru/article/n/logicheskoe-proektirovanie-relyatsionnoy-bazy-dannyh-na-osnove-printsipov-normalizatsii>

14. Гуляев П. С., Минакова О. В. Опыт проектирования API расписания мероприятий университета // Автоматика и программная инженерия 2023. № 1. С. 139-145. URL: <https://cyberleninka.ru/article/n/opyt-proektirovaniya-api-raspisaniya-meropriyatiy-universiteta>

15. Наумова Е. М., Метелев В. А., Ксенофонтова О. Л., Смирнова Н. В. Разработка веб-приложения «МедБлокнот: здоровье в цифрах» на стеке PostgreSQL, TypeScript, React // Современные наукоемкие технологии. Региональное приложение 2024. № 4. С. 121-129. URL: <https://cyberleninka.ru/article/n/razrabotka-veb-prilozheniya-medbloknot-zdorovie-v-tsifrah>

List of references

1. Regulations on Practical Training of Students: regulation. Ukhta: USTU, 2020. 25 p.
2. Rochev K.V. Information Technologies. Analysis and Design of Information Systems: textbook. Ukhta: USTU, 2018. 139 p.
3. Ilyushechkin V.M. Fundamentals of Database Use and Design. Moscow: Yurait, 2013. 224 p.
4. Rochev K. V. Classification of Graphical Modeling Tools for Information Systems Development // Information Technologies in Management and Economics. 2024. No. 1. Available at: <http://itue.ru/Issue/Article/275>.
5. Node.js Documentation. Available at: <https://nodejs.org/docs/latest/> (accessed 10.12.2024).
6. React Documentation. Available at: <https://ru.legacy.reactjs.org/docs/> (accessed 10.12.2024).
7. Mazarchuk M.S., Tyzhnenko Yu.V. Automated information system "Dekanat". Object Systems. 2011. No. 1. Pp. 53–58. Available at: <https://cyberleninka.ru/article/n/avtomatizirovannaya-informatsionnaya-sistema-dekanat>
8. Matyushenko V.A., Filon V.Yu., Beloded N.I. Information system for automating the preparation of educational process documents. System Analysis and Applied Informatics. 2016. No. 2. Pp. 66–70. Available at: <https://cyberleninka.ru/article/n/informatsionnaya-sistema-avtomatizatsii-podgotovki-dokumentov-uchebnogo-protssessa>
9. Astakhova T.N., Berdnikova A.A., Benben D.V., Lyamin A.S. Automation of accounting of scientific and educational-methodical activity of a teacher. International Journal of Open Information Technologies. No. 2. Pp. 117–125. Available at: <https://cyberleninka.ru/article/n/avtomatizatsiya-ucheta-nauchnoy-i-uchebnometodicheskoy-deyatelnosti-prepodavatelya>
10. Faizrahmanova K.E. Service-oriented architecture and technological implementation of a web service. Bulletin of Science. 2026. No. 4. Pp. 812–818. Available at: <https://cyberleninka.ru/article/n/servis-orientirovannaya-arhitektura-i-tehnologicheskaya-realizatsiya-veb-servisa>
11. Sergacheva M.A., Mikhalevskaya K.A. Analysis of frameworks for developing modern web applications. Kronos: Natural and Technical Sciences. 2020. No. 2. Pp. 35–39. Available at: <https://cyberleninka.ru/article/n/analiz-freymvorkov-dlya-razrabotki-sovremennyh-veb-prilozheniy>
12. Khairullin D.A., Bogdanov M.R. Studying the use of TypeScript in React applications, advantages of static typing. International Journal of Humanities and Natural Sciences. 2024. No. 12-3. Pp. 233–237. Available at: <https://cyberleninka.ru/article/n/izuchenie-ispolzovaniya-typescript-v-react-prilozheniyah-preimuschestva-staticheskoy-tipizatsii-podhody-k-tipizatsii-komponentov>
13. Ignatyuk V.A., Storozhok E.A. Logical design of a relational database based on normalization principles. Territory of New Opportunities. Bulletin of Vladivostok

State University. 2011. No. 2. Pp. 101–105. Available at: <https://cyberleninka.ru/article/n/logicheskoe-proektirovanie-relyatsionnoy-bazy-dannyh-na-osnove-printsipov-normalizatsii>

14. Gulyaev P.S., Minakova O.V. Experience in designing an API for a university event schedule. Automation and Software Engineering. 2023. No. 1. Pp. 139–145. Available at: <https://cyberleninka.ru/article/n/opyt-proektirovaniya-api-raspisaniya-meropriyatiy-universiteta>

15. Naumova E.M., Metelev V.A., Ksenofontova O.L., Smirnova N.V. Development of a web application "MedBloknot: health in numbers" on the PostgreSQL, TypeScript, React stack. Modern Science-Intensive Technologies. Regional Supplement. 2024. No. 4. Pp. 121–129. Available at: <https://cyberleninka.ru/article/n/razrabotka-veb-prilozheniya-medbloknot-zdorovie-v-tsifrah>