

УДК 004.4

Разработка информационной системы автоматизированного учета книгооборота в библиотеках

Мусаев Максим Шахинович¹

Никитцев Антон Валерьевич²

Волосова Александра Владимировна¹ (научный руководитель)

¹ МГТУ им. Н.Э. Баумана, Москва, Россия

² МАИ, Москва, Россия

Статья посвящена разработке автоматизированной системы учета книгооборота для библиотек. Система построена на клиент-серверной архитектуре: мобильное (Flutter) и десктопное (Flutter) приложения для читателей и библиотекарей, серверная часть — Node.js с PostgreSQL (основные данные) и Redis (кеш). Модули: пользователь, библиотекарь, админ-панель. Автоматизация операций реализована через QR-коды (идентификация читателей) и штрих-коды (учет книг). Цель — автоматизация выдачи/возврата книг, контроля задолженностей и книгооборота, сокращение трудозатрат и ошибок. Кроссплатформенность, высокая производительность и безопасность (шифрование, аутентификация) обеспечены современными технологиями. Тестирование подтвердило ускорение процессов и повышение точности учета. Внедрение системы оптимизирует работу библиотек, улучшит обслуживание и адаптируемость под различные учреждения.

Ключевые слова: автоматизация, библиотечный учет, книгооборот, программное обеспечение, информационные системы, монолитные системы

Введение. Несмотря на активное развитие онлайн-библиотек, большое количество людей, в особенности школьники и студенты, отдает предпочтение бумажной литературе, однако деятельность библиотек часто осложняется устаревшими методами учета книгооборота. Ручная регистрация выдачи и возврата книг, составление отчетов и управление фондами требуют значительных временных и трудовых затрат, что особенно заметно в учреждениях с большим числом посетителей. Эти процессы не только затрудняют работу сотрудников, но и создают неудобства для читателей.

Автоматизация библиотечных процессов является актуальным решением этих проблем, позволяя библиотекам повысить качество обслуживания, сократить нагрузку на персонал и минимизировать ошибки. Информационные системы, специально разработанные для автоматизации учета, способны существенно улучшить организацию библиотечных процессов, используя современные инструменты для управления фондами и обработки данных о книгообороте.

Статья посвящена разработке автоматизированной информационной системы учета книгооборота, ориентированной на библиотеки различных типов. Основное внимание уделено проектированию и реализации программно-

го продукта, способного адаптироваться под особенности учебных, муниципальных и корпоративных библиотек.

Обзор аналогов. На рынке автоматизации библиотек существует множество решений, каждое из которых направлено на улучшение управления книгооборотом и упрощение работы сотрудников. Среди наиболее популярных и общедоступных систем в России можно выделить такие продукты, как ИРБИС64 [1], МЕГАПРО [2] и МАРК-SQL [3].

ИРБИС64 [1] представляет собой модульную автоматизированную библиотечно-информационную систему, которая может быть использована даже на компьютерах с устаревшими операционными системами. Основное преимущество этой системы заключается в низких системных требованиях, что делает ее доступной для многих организаций. Однако устаревшая архитектура накладывает ограничения на возможности интеграции с современными технологиями, а также может создавать риски в области информационной безопасности.

Система МЕГАПРО [2] также имеет модульную структуру, но отличается использованием веб-технологий, что позволяет ей работать как в локальных сетях, так и через интернет. Это делает ее более универсальной по сравнению с ИРБИС64. Тем не менее, зависимость от продуктов Microsoft, таких как IIS и .NET Framework, может стать препятствием для долгосрочной поддержки, если эти технологии перестанут развиваться.

МАРК-SQL [3] предлагает высокую гибкость в выборе баз данных и обеспечивает интеграцию с различными информационными системами. Однако она предъявляет значительные требования к вычислительным ресурсам, что делает ее менее доступной для организаций с ограниченным бюджетом. Кроме того, ее использование предполагает наличие мощного сервера и достаточно квалифицированного персонала для администрирования.

В рамках исследования был учтен опыт указанных решений, что позволило разработать систему, лишенную их основных недостатков. Новая система интегрирует современные технологии для автоматической идентификации пользователей и взаимодействия с фондами, сохраняя при этом высокую производительность и адаптивность к различным библиотечным условиям.

Методы разработки. Процесс создания автоматизированной системы учета книгооборота начался с анализа потребностей библиотек и определения функциональных требований к системе. Были рассмотрены ключевые аспекты, такие как ускорение процесса выдачи и возврата книг, возможность точного учета задолженностей, обеспечение доступности информации о фондах для читателей и сотрудников, а также интеграция современных технологий идентификации пользователей (рис. 1).

На основе выявленных требований была спроектирована архитектура системы, состоящая из двух основных модулей: модуль библиотекаря и модуль пользователя. Также в backend-части была добавлена функциональность администрирования системы. Такая структура позволяет адаптировать функци-

онал под задачи каждой из категорий пользователей. Модуль библиотекаря предназначен для операций взаимодействия с книгами и пользователями, а модуль пользователя дает возможность получать информацию о доступных книгах, бронировать их и следить за сроками возврата.

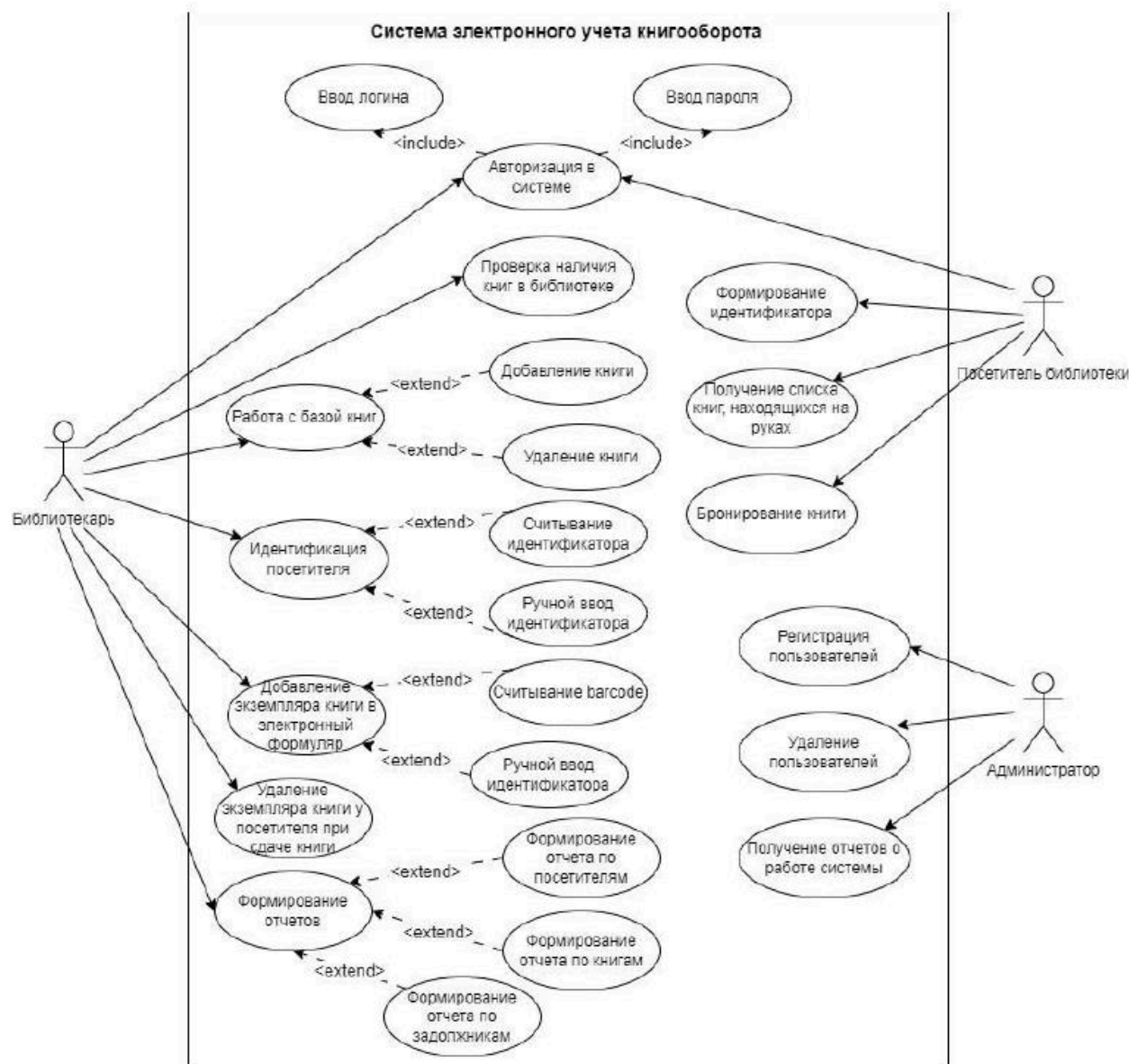


Рис. 1. Диаграмма вариантов использования

Техническая реализация системы опиралась на современные инструменты и технологии. В качестве языка программирования для клиентской части использовался Dart [4] с использованием фреймворка Flutter [5], что обеспечило создание кроссплатформенного интерфейса. Для серверной части был выбран Node.js [6] в сочетании с Express [7]. Для управления данными использовалась база данных PostgreSQL [8] с ORM Prisma [9], а для кеширования использовался Redis [10]. В совокупности эти решения позволили обеспечить высокую производительность и гибкость работы с данными.

Проектирование интерфейса выполнялось с учетом удобства и интуитивной понятности для конечных пользователей. Например, модуль библиотекаря включает такие функции, как автоматический поиск книг по штрих-коду или пользователя по идентификатору, просмотр задолженностей, а также возможность формирования отчетов. Пользователи, в свою очередь, могут с помощью мобильного приложения просматривать доступные книги и бронировать их, получать уведомления о сроках возврата, а также оставлять отзывы на прочитанные книги. Отдельное внимание было уделено интеграции технологий идентификации. Использование QR-кодов для читателей и штрих-кодов для книг позволяет значительно ускорить процессы выдачи и возврата, минимизируя человеческий фактор. Для хранения и обработки персональных данных соблюдены требования законодательства [11], что обеспечило безопасность и конфиденциальность информации.

Результатом этапа проектирования и реализации стала полнофункциональная система, адаптируемая под различные условия эксплуатации.

Реализация продукта. Создание автоматизированной системы учета книгооборота включало несколько этапов: проектирование, создание дизайна, реализация frontend части, реализация backend части. После данных этапов команда получила MVP (Минимально жизнеспособный продукт) версию продукта, которая уже после нескольких циклов получения обратной связи и обновления стала финальной.

В первой части разработки было проведено проектирование базы данных (рис. 2). Для хранения информации о книгах, пользователях, операциях с фондами и задолженностях была использована реляционная база данных PostgreSQL [8]. Для упрощения взаимодействия с базой данных был выбран ORM Prisma [9], что позволило ускорить разработку и обеспечило высокую производительность запросов.

На основе этой структуры была реализована серверная часть системы с использованием фреймворка Express [7] для Node.js [6]. Сервер обеспечивает обработку запросов от пользователей и передачу данных в базу, а также предоставляет API для взаимодействия с клиентскими приложениями. Серверная часть обеспечивает работу всех ключевых функций системы, таких как регистрация пользователей, а также обработка заявок на книги.

Клиентская часть системы была разработана с использованием Dart [4] и фреймворка Flutter [5], что позволило создать быстрое кроссплатформенное мобильное приложение для пользователей библиотеки. Интерфейс приложения включает несколько ключевых экранов (рис. 3).

Главная страница — доступ к списку книг, поиск и фильтрация по различным критериям (автор, жанр, доступность).

Информация о книге — отображение подробных данных о выбранной книге, возможность забронировать книгу или добавить в список желаемого.

Профиль — информация о пользователе, список взятых книг, задолженности, возможность продления сроков возврата.

Заказы — список активных и завершенных заказов и даты их сдачи.

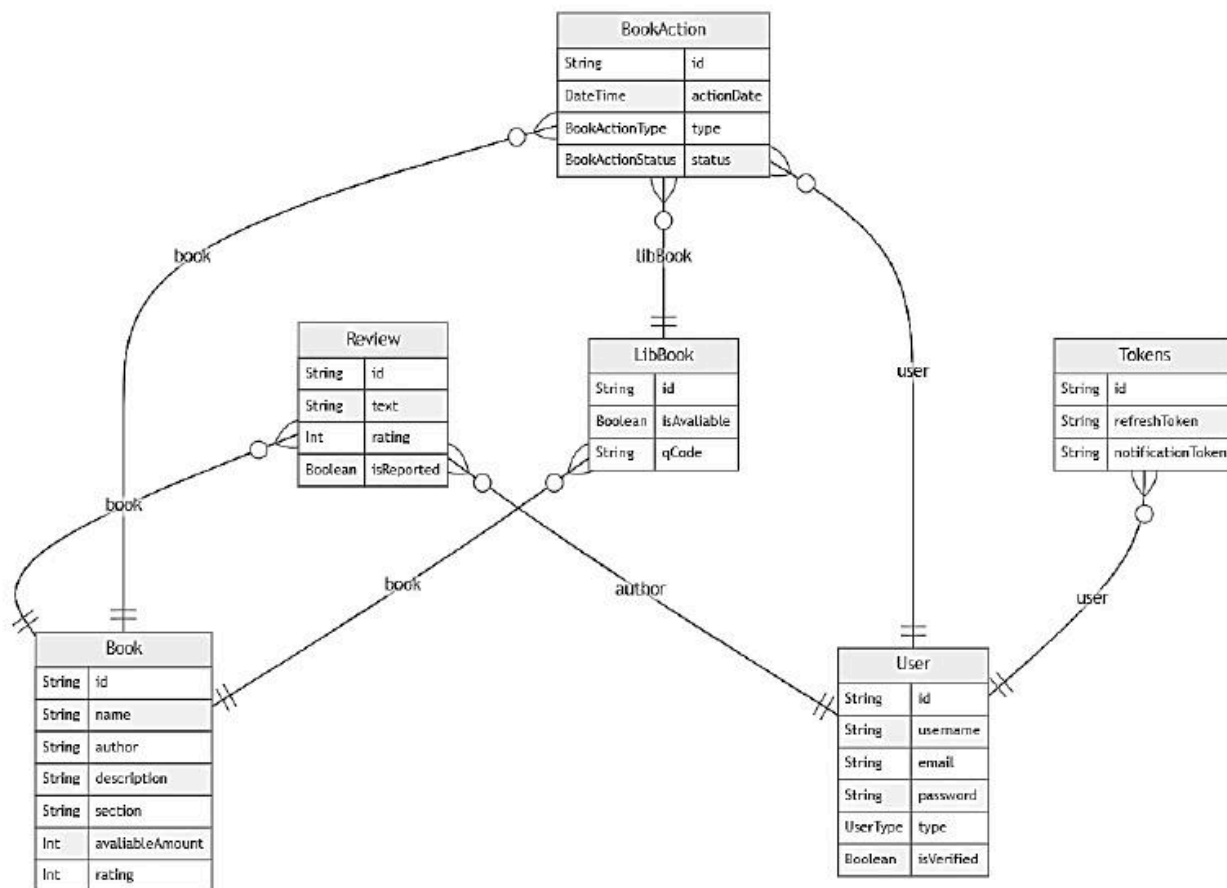


Рис. 2. ER-диаграмма базы данных

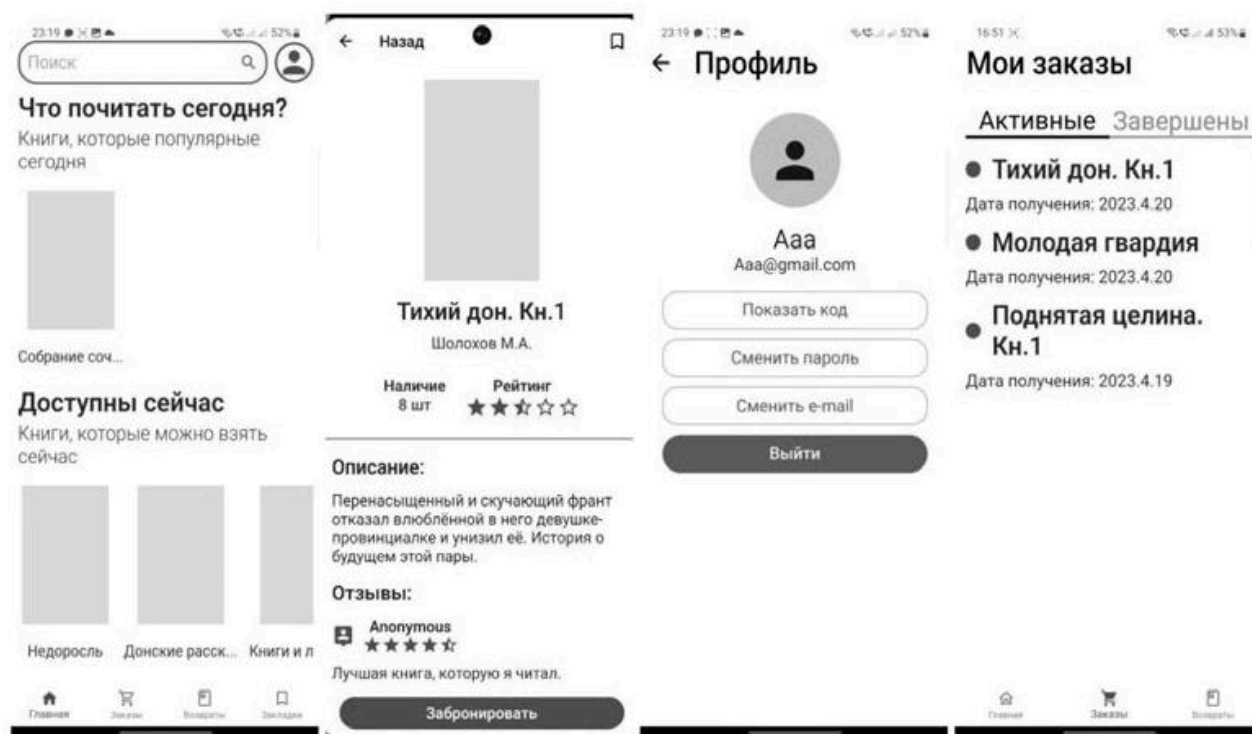


Рис. 3. Ключевые экраны клиентской версии

Модуль библиотекаря включает функциональность для быстрого поиска книг и пользователей, регистрации выдачи и возврата, а также учета задолженностей (рис. 4). Для удобства работы предусмотрено сканирование QR-кодов и штрих-кодов, что существенно ускоряет процессы и минимизирует человеческий фактор. Библиотекарь может видеть всю информацию о книгах, возвращать или продлевать срок их выдачи, а также просматривать статистику платформы.

Одним из ключевых этапов реализации было создание системы для автоматической идентификации пользователей. Для этого используется технология QR-кодов, которые генерируются в мобильном приложении пользователя. Библиотекарь может сканировать код с экрана мобильного устройства и мгновенно получить доступ к данным о пользователе, что ускоряет процесс выдачи книг и исключает ошибки.

Для повышения безопасности и защиты персональных данных была предусмотрена система аутентификации, которая использует шифрование паролей и защиту данных с помощью современных методов криптографии. Все данные пользователей хранятся в зашифрованном виде, а доступ к информации строго контролируется.



Рис. 4. Основной экран библиотекаря

Реализованная система имеет высокую степень адаптивности и позволяет легко интегрировать дополнительные модули, такие как отчетность, анализ статистики или любые другие внешние модули.

Результаты и тестирование. Тестирование разработанной системы было направлено на проверку корректности выполнения всех ключевых функ-

ций, устойчивости при нагрузке и удобства взаимодействия для конечных пользователей. Проверялись все основные сценарии — от выдачи и возврата книг до автоматической идентификации через QR-коды.

Полученные результаты показали следующие преимущества.

1. Скорость обработки операций увеличилась примерно на 60 % по сравнению с ручными методами. Использование QR-кодов и автоматизация ввода данных позволили значительно ускорить процессы обслуживания.

2. Снижение количества ошибок достигло 60 % — автоматическая проверка корректности данных и исключение ручного ввода обеспечили высокую точность учета.

3. Стабильная производительность наблюдалась при параллельной работе нескольких пользователей. Все основные действия выполнялись без ощутимых задержек, включая загрузку списков книг, бронирование и оформление возврата, что свидетельствует о надежности системы при реальной нагрузке.

4. Надежность и безопасность данных обеспечены средствами шифрования и разграничением прав доступа. В ходе тестирования не было зафиксировано случаев несанкционированного доступа или потери информации.

5. Положительная обратная связь получена от пользователей: более 83 % респондентов отметили простоту интерфейса, интуитивную навигацию и общее повышение качества и скорости обслуживания.

Результаты тестирования подтверждают, что система готова к внедрению и способна существенно повысить эффективность работы библиотек.

Заключение. По итогам выполненной работы можно сделать следующие выводы. Разработана и протестирована автоматизированная система учета книгооборота, которая значительно улучшает эффективность работы библиотеки и повышает удобство взаимодействия с пользователями. Автоматическая идентификация пользователей через QR-коды, позволяет ускорить процессы выдачи и возврата книг, а также снизить нагрузку на сотрудников.

Использование реляционной базы данных PostgreSQL [8] и кроссплатформенного интерфейса Flutter [5] обеспечивает высокую производительность и удобство использования как для библиотекарей, так и для читателей.

Тестирование показало, что система успешно выполняет все ключевые функции, включая обработку данных о книгообороте, учет задолженностей и управление библиотечными фондами. Проводимые операции происходят быстро и безошибочно, что позволяет существенно сэкономить время, которое раньше тратилось на ручной учет.

Система обеспечивает высокий уровень безопасности данных, что является важным аспектом при работе с персональной информацией пользователей. Все данные защищены с помощью современных методов шифрования, а механизмы аутентификации и авторизации гарантируют конфиденциальность и целостность данных.

Созданная система представляет собой удобное и эффективное решение для автоматизации учета книгооборота в библиотеках, и ее внедрение прине-

сет значительные преимущества в работе как небольших, так и крупных библиотек.

Перспективы дальнейшего развития системы включают создание отдельного интерфейса для администратора или его интеграцию в существующее десктопное приложение и внедрение дополнительных технологий, таких как NFC для идентификации пользователей, что позволит еще больше ускорить процессы учета. Также планируется интеграция с внешними сервисами для автоматической обработки данных о книгах, что облегчит процесс добавления новых материалов в каталог.

Литература

- [1] *ИРБИС64: Автоматизированная библиотечная информационная система*. URL: https://webirbis.tsogu.ru/cgi-bin/irbis64r_plus/cgiirbis_64_ft.exe (дата обращения 16.12.2024).
- [2] *МЕГАПРО: Система автоматизации библиотек*. URL: <https://www.megaprosoft.ru> (дата обращения 16.12.2024).
- [3] *МАРК-SQL: Автоматизация библиотечных процессов*. URL: <https://www.informsystema.ru/content/АИБС-семейства-МАРК-sql> (дата обращения 16.12.2024).
- [4] *Dart: официальная документация*. URL: <https://dart.dev/docs> (дата обращения 16.12.2024).
- [5] *Flutter: официальная документация*. URL: <https://docs.flutter.dev> (дата обращения 16.12.2024).
- [6] *Node.js: официальная документация*. URL: <https://nodejs.org/docs/latest/api> (дата обращения 16.12.2024).
- [7] *Express.js: официальная документация*. URL: <https://expressjs.com/ru> (дата обращения 16.12.2024).
- [8] *PostgreSQL: официальная документация*. URL: <https://www.postgresql.org/docs> (дата обращения 16.12.2024).
- [9] *Prisma ORM: официальная документация*. URL: <https://www.prisma.io/docs> (дата обращения 16.12.2024).
- [10] *Redis: официальная документация*. URL: <https://redis.io/docs> (дата обращения 16.12.2024).
- [11] *О персональных данных. Федеральный закон*. URL: <http://pravo.gov.ru/proxy/ips/?docbody&nd=102108261> (дата обращения 16.12.2024).

Development of an information system for automated library circulation management

Musaev Maxim Shahinovich¹

Nikittsev Anton Valerievich²

Volosova Alexandra Vladimirovna¹ (*scientific advisor*)

¹ *BMSTU, Moscow, Russia*

² *MAI, Moscow, Russia*

The article presents the development of an automated book circulation management system for libraries. The system employs client-server architecture: a mobile (Flutter) app for patrons, a cross-platform desktop (Flutter) app for librarians, and a Node.js backend with

PostgreSQL (core data) and Redis (caching). Modules include patron, librarian, and admin panel functionalities. Automation utilizes QR codes (patron identification) and barcodes (book tracking). The system aims to automate book check-in/check-out, debt management, and circulation control while reducing manual effort and errors. Modern technologies ensure cross-platform compatibility, high performance, and security (encryption, authentication). Testing confirmed accelerated operations and improved accounting accuracy. Implementation optimizes library workflows, enhances user service, and ensures adaptability for diverse institutions.

Keywords: automation, library accounting, circulation management, software, information systems, monolithic systems