

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической кибернетики и компьютерных наук

**ПРИЛОЖЕНИЕ ДЛЯ АНАЛИЗА И ФОРМИРОВАНИЯ ОТЧЕТОВ ПО
МЕДИЦИНСКИМ ЛАБОРАТОРНЫМ ИССЛЕДОВАНИЯМ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 411 группы

направления 02.03.02 — Фундаментальная информатика и информационные
технологии

факультета КНиИТ

Долесова Валентина Пантелеевича

Научный руководитель

доцент, к. ф.-м. н.

М. И. Сафрончик

Заведующий кафедрой

к. ф.-м. н., доцент

С. В. Миронов

Саратов 2026

ВВЕДЕНИЕ

Лабораторная диагностика остаётся одним из основных источников объективной информации о состоянии пациента и применяется на всех этапах лечебного процесса. Объём выполняемых исследований устойчиво растёт, что увеличивает нагрузку на медицинский персонал, ответственный за интерпретацию и систематизацию результатов.

Тем не менее во многих медицинских организациях обработка результатов анализов по-прежнему ведётся вручную или с помощью разрозненных программных средств общего назначения — электронных таблиц и текстовых редакторов. Это повышает риск ошибок при переносе данных, затрудняет сравнение показателей в динамике и не обеспечивает единого формата хранения и представления информации.

Устранить эти недостатки способны специализированные информационные системы, которые автоматически принимают лабораторные данные, сравнивают их с медицинскими нормами и формируют структурированные отчёты с доступом к информации через мобильные устройства. Настоящая работа посвящена проектированию и реализации такой системы.

Цель работы — разработка архитектуры и прототипа приложения для автоматизированного анализа медицинских лабораторных исследований и формирования отчётов с доступом к данным через мобильное приложение.

Для достижения поставленной цели необходимо решение следующих задач:

1. проанализировать предметную область лабораторной диагностики и существующие программные решения;
2. сформулировать функциональные и нефункциональные требования к приложению;
3. исследовать современные технологии и архитектурные подходы к построению медицинских информационных систем;
4. спроектировать архитектуру приложения из серверной части и мобильного клиента;
5. разработать прототип серверной части для загрузки, анализа и хранения лабораторных данных;
6. создать мобильное приложение для просмотра результатов анализов и отчётов.

Структура и объём работы. Бакалаврская работа состоит из введения, двух разделов, заключения, списка использованных источников и пяти приложений. Общий объём работы — 62 страницы, из них 39 страниц — основное содержание, включающее 17 рисунков; список использованных источников информации — 20 наименований; в качестве приложения работа дополнена цифровым носителем.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Анализ предметной области и выбор средств реализации решения» посвящён анализу предметной области лабораторной диагностики, обзору существующих программных решений, формированию требований к системе и обоснованию выбора технологических средств и архитектуры приложения.

В разделе рассмотрены классы программных систем автоматизации работы с медицинскими данными — медицинские, госпитальные и лабораторные информационные системы, а также облачные порталы лабораторий. Анализ показал, что промышленные системы избыточны для узкой задачи анализа результатов, их логика интерпретации обычно ограничена проверкой референсного интервала, а сами правила не поддаются настройке силами медицинского персонала. На основании этого обоснована ниша для специализированного приложения, ключевыми отличиями которого являются настраиваемая база клинических правил, формирование заключений с прослеживаемой логикой вывода и мобильный клиент как основной интерфейс взаимодействия.

Сформулированы функциональные и нефункциональные требования к приложению. К функциональным отнесены: регистрация и аутентификация пользователей с разграничением прав по ролям врача, пациента и администратора; загрузка файлов лабораторных исследований в формате электронных таблиц с автоматическим разбором содержимого; сопоставление показателей со справочником медицинских норм; применение настраиваемой базы клинических правил с формированием обоснованных заключений; хранение истории отчётов; ведение справочника норм и базы правил; обмен сообщениями между пациентом и врачом; экспорт отчётов в формат PDF и управление учётными записями. К нефункциональным требованиям отнесены безопасность хранения и передачи данных, кроссплатформенность, расширяемость, воспроизводимость развёртывания, надёжность и достаточная производительность.

Выполнен сравнительный анализ технологических средств. В качестве платформы серверной части из рассмотренных вариантов выбрана связка Java 17 и Spring Boot 3, обеспечивающая статическую типизацию, развитую экосистему и высокую пропускную способность. Клиентская часть реализована на кроссплатформенном фреймворке Flutter с языком Dart, в качестве СУБД выбрана PostgreSQL с поддержкой типа JSONB и инструментом версионирования схемы

Liquibase, развёртывание организовано средствами Docker и Docker Compose. Спроектирована общая архитектура приложения, выделены два независимых компонента — серверное приложение, реализующее бизнес-логику и экспертную подсистему, и мобильный клиент, выступающий основным интерфейсом для всех категорий пользователей; взаимодействие между компонентами организовано через REST-интерфейс с использованием формата JSON.

Второй раздел «Реализация приложения» посвящён практической реализации спроектированной архитектуры: серверной части, модели данных и клиентского приложения.

Серверная часть структурирована по принципам многослойной архитектуры с разделением на слои контроллеров, сервисов, репозитория, моделей предметной области, объектов передачи данных, а также конфигурации и безопасности. Реализовано пятнадцать REST-контроллеров, объединённых в функциональные группы (аутентификация, обработка лабораторных данных, аналитика, справочник норм, чат, администрирование), а документация API автоматически формируется средствами OpenAPI и доступна в виде Swagger UI. Ключевой особенностью серверной части является экспертная подсистема, автоматически применяющая к каждому пациенту набор формализованных клинических правил. Условие срабатывания правила описывается компактным декларативным языком на основе JSON с логическими группами (И, ИЛИ, отрицание) и набором операторов сравнения; класс сборки фактов формирует структуру из примерно 35 признаков пациента, включая значения маркеров до и после операции, классификацию по системе TNM и дельты показателей между визитами, а интерпретатор условий рекурсивно вычисляет дерево условия. Все срабатывания правил сохраняются в журнале выполнения с указанием совпавших фактов и обоснований, что обеспечивает важное для медицинских систем свойство объяснимости заключений. Защита REST API реализована средствами Spring Security с аутентификацией на основе JWT, хешированием паролей алгоритмом BCrypt, ролевым доступом и ограничением частоты обращений к чувствительным эндпоинтам.

Спроектирована и реализована реляционная модель данных в СУБД PostgreSQL, включающая таблицы пользователей, реестра пациентов, справочника норм, отчётов и их строк, результатов аналитики, источников доказательной медицины, клинических правил, журнала их срабатываний, а также тредов и со-

общений чата. Тело отчёта, условия и результаты правил хранятся в формате JSONB, что сочетает гибкость документной модели с надёжностью реляционной СУБД. Управление схемой базы данных и её эволюцией выполняется средствами Liquibase в виде набора версионированных YAML-миграций, доступ к данным организован через Spring Data JPA с автоматической генерацией запросов по именам методов репозитория и параметризацией, исключающей SQL-инъекции на уровне фреймворка.

Клиентская часть разработана как мобильное приложение на платформе Flutter под Android. Состояние приложения распределено по провайдерам аутентификации, отчётов, экспертной системы и чата, транспортный слой инкапсулирован в едином клиенте на основе пакета http, а JWT-токен хранится в зашифрованном защищённом хранилище. Реализованы экраны аутентификации, главного экрана с адаптацией под роль пользователя, загрузки лабораторных файлов, списка и детального просмотра отчётов с цветовой индикацией отклонений и перечнем сработавших правил, чата между пациентом и врачом, а также административные экраны управления пользователями, реестром пациентов и базой клинических правил. Экспорт отчёта в PDF выполняется на сервере средствами Apache PDFBox, после чего файл открывается во встроенном системном просмотрщике устройства. Подробно рассмотрен сквозной сценарий загрузки лабораторного файла врачом, демонстрирующий совместную работу слоёв: прохождение фильтров безопасности, разбор файла, формирование отчёта, запуск экспертной системы и возврат агрегированной сводки клиенту.

В результате выполнения второго раздела получен полностью функциональный прототип приложения, готовый к развёртыванию.

ЗАКЛЮЧЕНИЕ

В дипломной работе разработано клиент-серверное приложение для автоматизированного анализа медицинских лабораторных исследований и формирования отчётов: серверная часть на Java/Spring Boot 3 и мобильный клиент на Flutter.

В теоретической части проведён анализ предметной области лабораторной диагностики и существующих программных решений, сформулированы функциональные и нефункциональные требования к приложению. Систематизированы вопросы безопасности персональных данных, многослойной архитектуры и продукционных экспертных систем; на основании сравнительного анализа обоснован выбор стека Java 17 + Spring Boot 3 + PostgreSQL + Liquibase + Flutter.

В практической части спроектирована архитектура приложения, разработан прототип серверной части и создано мобильное приложение. Получены следующие основные результаты:

1. Многослойная серверная архитектура (контроллеры, сервисы, репозитории), изолирующая бизнес-логику и упрощающая замену компонентов.
2. Экспертная подсистема на настраиваемых правилах: декларативное описание условий с учётом динамики показателей пациента, редактирование администратором без перекомпиляции и журнал срабатываний с обоснованиями обеспечивают аудит и объяснимость заключений.
3. Защищённый REST API (JWT-аутентификация, хеширование паролей, ролевой доступ, ограничение частоты запросов), автоматическая API-документация, версионирование схемы базы данных миграциями и развёртывание через Docker.
4. Мобильный клиент на Flutter под Android с поддержкой ролей врача, пациента и администратора; реализованы загрузка и разбор лабораторных файлов, экспорт PDF-отчётов, чат пациент–врач и административное управление пользователями, нормами и базой правил.

Практическая значимость разработанного приложения подтверждена внедрением результатов дипломной работы в учебный процесс Медицинского университета «Реавиз». Согласно акту о внедрении от 1 июня 2026 г., результаты исследования, посвящённые анализу и формированию отчётов по медицинским лабораторным исследованиям, используются при чтении лекций и проведении практических занятий на кафедре хирургических болезней со студентами и

слушателями ФПК и ППС.

Кроме того, разработанное программное решение прошло государственную регистрацию в Федеральной службе по интеллектуальной собственности. На программу для ЭВМ «Приложение для анализа и формирования отчетов по медицинским лабораторным исследованиям» получено свидетельство о государственной регистрации № 2026664629 от 18 мая 2026 г. Наличие акта о внедрении и охранного документа подтверждает прикладной характер выполненной работы, её актуальность для образовательной и медицинской сферы, а также возможность дальнейшего развития системы и её применения в практической деятельности.

Таким образом, цель работы — разработка архитектуры и прототипа приложения для автоматизированного анализа медицинских лабораторных исследований и формирования отчётов с доступом через мобильное приложение достигнута, а поставленные задачи решены в полном объёме.