

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

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

Кафедра математической кибернетики и компьютерных наук

**РАЗРАБОТКА И РАЗВЕРТЫВАНИЕ СЕРВЕРНОЙ ЧАСТИ ДЛЯ
АНАЛИЗА РЕЗЮМЕ НА ОСНОВЕ МЕТОДОВ ИСКУССТВЕННОГО
ИНТЕЛЛЕКТА**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 411 группы

направления 02.03.02 — Фундаментальная информатика и информационные
технологии

факультета КНиИТ

Пшеничникова Стемира Михайловича

Научный руководитель

доцент, к. ф.-м. н.

М. И. Сафрончик

Заведующий кафедрой

к. ф.-м. н., доцент

С. В. Миронов

Саратов 2026

ВВЕДЕНИЕ

Актуальность темы. В современном мире профессиональный отбор и оценка кандидатов являются важными задачами для компаний, стремящихся эффективно находить подходящих сотрудников. Ручной анализ резюме занимает много времени и может быть субъективным. Поэтому автоматизация процесса оценки компетенций с использованием методов машинного обучения и обработки естественного языка становится актуальной задачей.

Данная работа выполнена в рамках проекта «HR-Интеллект», целью которого является создание программного комплекса для автоматизированного анализа резюме, оценки компетенций кандидатов и их сопоставления с требованиями целевых должностей. Система предназначена для решения задач первичного скрининга в HR-отделах компаний, позволяя сократить временные затраты рекрутеров и систематизировать процесс подбора персонала. Проект реализуется в формате «Стартап как диплом», что предполагает не только теоретическую проработку, но и создание готового к внедрению программного продукта.

В условиях современного рынка труда эффективный подбор персонала является одним из ключевых факторов конкурентоспособности компаний. Рекрутер вынужден просматривать десятки и сотни резюме, сопоставляя описанный опыт кандидатов с требованиями вакансий. Помимо временных издержек, ручной анализ подвержен субъективности: оценка компетенций может варьироваться в зависимости от опыта и квалификации конкретного специалиста, а формулировки навыков в резюме часто не совпадают с терминологией, принятой в компании. В этой связи автоматизация процесса оценки компетенций с использованием методов машинного обучения и обработки естественного языка становится актуальной научно-практической задачей, решение которой позволяет кратно сократить время первичного скрининга при одновременном повышении объективности и стандартизации принимаемых кадровых решений.

Цель бакалаврской работы – разработка и промышленное развертывание серверной части программного комплекса «HR-Интеллект», предназначенного для автоматизированной оценки компетенций кандидатов на основе анализа их резюме с использованием методов искусственного интеллекта.

Поставленная цель определила следующие задачи:

1. Спроектировать архитектуру системы и схемы взаимодействия между

клиентским приложением, серверным API, базой данных и ML-моделями.

2. Разработать структуру базы данных для хранения информации о пользователях, профессиях и компетенциях.

3. Реализовать серверную часть на FastAPI с RESTful API для аутентификации пользователей и управления данными.

4. Контейнеризовать компоненты системы с использованием Docker и настроить межсервисное взаимодействие.

5. Обеспечить отказоустойчивое развертывание системы в производственной среде.

Методологические основы разработки серверной части распределённых программных комплексов с применением методов искусственного интеллекта представлены в работах С. Ньюмена в области архитектуры микросервисов, В. Винсента в области веб-разработки на Python, Дж. Майерса и Р. Коупленда в области проектирования баз данных с использованием ORM. Проектирование REST API осуществлялось с опорой на существующие стандарты и руководства. Техническая реализация базировалась на официальной документации используемых технологий: FastAPI, SQLAlchemy, Alembic, Docker.

Теоретическая значимость бакалаврской работы заключается в систематизации подходов к проектированию распределённых систем анализа текстовой информации с использованием микросервисной архитектуры, обосновании выбора JSONB как эффективного типа данных для хранения слабоструктурированных матриц компетенций в реляционных СУБД, а также в разработке модели многоэтапного пайплайна обработки резюме, сочетающего семантический поиск и контекстную классификацию.

Практическая значимость работы состоит в создании готового к внедрению программного продукта, обеспечивающего автоматизацию первичного скрининга резюме в HR-отделах компаний. Разработанный программный комплекс позволяет сократить временные затраты рекрутеров на анализ резюме, повысить объективность оценки кандидатов за счёт применения методов машинного обучения и стандартизировать процесс сопоставления компетенций с требованиями вакансий. Контейнеризация всех компонентов и настройка оркестрации через Docker Compose обеспечивают возможность развёртывания системы в различных производственных средах одной командой.

Структура и объём работы. Бакалаврская работа состоит из введения,

двух разделов, заключения, списка использованных источников и двух приложений. Общий объем работы – 66 страниц, из них 57 страницы – основное содержание, включая 3 рисунков и 10 таблиц, цифровой носитель в качестве приложения, список использованных источников информации – 20 наименований.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Архитектурное проектирование и моделирование данных программного комплекса» посвящен анализу предметной области, формированию требований к системе, проектированию архитектуры распределённого приложения и моделированию структуры базы данных. В разделе рассмотрены концепции и функциональная структура системы «HR-Интеллект», представляющей собой интеллектуальную систему обработки текстовой информации, реализующую полный цикл анализа: от загрузки исходного документа до получения интерпретируемого заключения. В основе системы лежит гибридный подход, сочетающий семантический поиск и контекстный анализ текста с применением методов обработки естественного языка. В отличие от классических алгоритмов фильтрации по ключевым словам, такой подход позволяет учитывать смысловую близость формулировок, взаимосвязи между навыками и уровень владения компетенциями.

Функциональная структура программного комплекса реализована в виде распределённой системы из трёх взаимодействующих микросервисов, образующих единый конвейер обработки данных. Первый этап обработки включает извлечение текстового содержимого из загруженного документа, очистку данных и приведение текста к унифицированному виду. Далее backend-сервис направляет подготовленный текст в ML-микросервис `ml-extractor`, реализующий семантический поиск на основе модели Sentence-BERT. Модель преобразует текст резюме и описания компетенций в векторные представления фиксированной размерности, после чего вычисляет косинусное сходство между ними. Полученные пары «компетенция-предложение» передаются в `ml-grader`, построенный на основе RuBERT и дообученный на специализированном датасете. Разделение ML-логики на два независимых сервиса обеспечивает соблюдение принципа единой ответственности, позволяет независимо масштабировать каждый сервис и изолированно обновлять модели.

В разделе выполнен анализ технологического стека и распределение ролей в команде проекта. Обоснован выбор Python и FastAPI для backend-сервиса, PostgreSQL в качестве СУБД, SQLAlchemy для ORM и Alembic для миграций. Проведено концептуальное моделирование предметной области с выделением ключевых сущностей: пользователь, компетенция, профессия, резюме, результат анализа. Построена ER-диаграмма базы данных (рисунок 1).

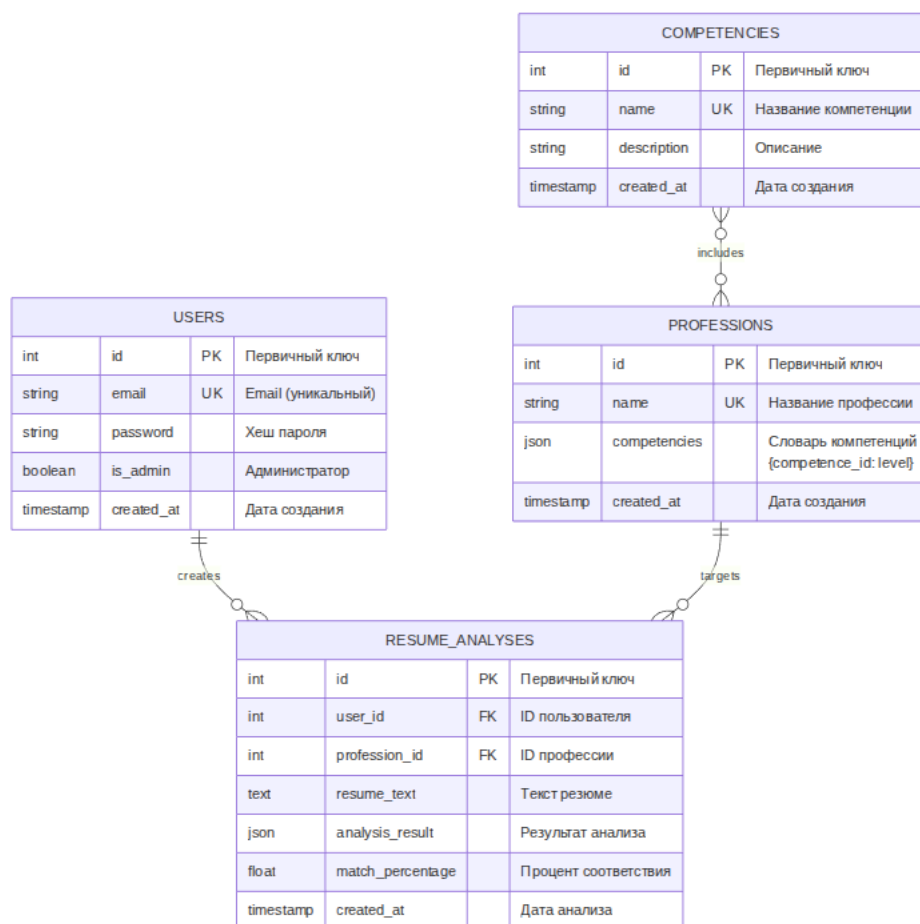


Рисунок 1 – ER-диаграмма базы данных программного комплекса «HR-Интеллект»

Спроектирована реляционная база данных в схеме `schema_competency`, включающая таблицы `users`, `competencies`, `professions`, `invitation_links`, `resumes` и `analysis_results`. Ключевым архитектурным решением является использование типа `JSONB` для хранения матриц компетенций и результатов анализа, что сочетает гибкость документоориентированного подхода с производительностью реляционной СУБД.

Детально спроектирована архитектура распределённой системы на основе микросервисного подхода, разработана UML-диаграмма компонентов (рисунок 2), а также диаграмма последовательности действий.

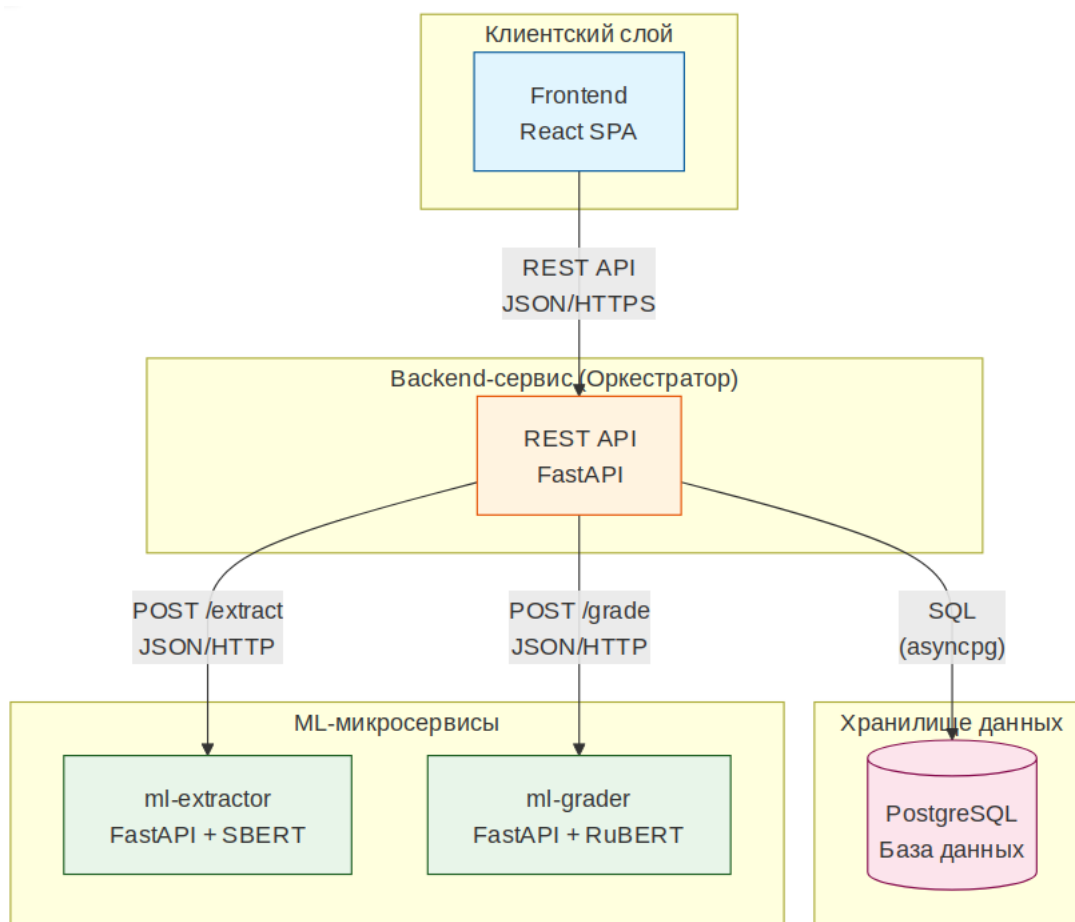


Рисунок 2 – UML-диаграмма компонентов программного комплекса «HR-Интеллект»

Определены интерфейсы взаимодействия: REST API backend-сервиса (аутентификация, CRUD-операции для компетенций и профессий, управление ссылками-приглашениями, анализ резюме) и внутренние API ML-микросервисов. Для каждой конечной точки определены HTTP-методы, форматы запросов и ответов, коды статусов и требования к аутентификации.

В результате выполнения первого раздела сформирована полная архитектурная и информационная модель системы, послужившая основой для практической реализации.

Второй раздел «Серверная реализация и инфраструктура программного комплекса» посвящен практической реализации спроектированной архитектуры. Автором самостоятельно разработаны все компоненты серверной части системы.

Реализована база данных на основе моделей SQLAlchemy в асинхронном режиме с драйвером asyncpg. Базовый класс моделей содержит карту аннотаций типов, автоматически сопоставляющую Python-типы с SQL-типами PostgreSQL. Поле `competencies` модели `Profession` типизировано как `dict[int, int]` на

уровне Python и JSONB на уровне базы данных, что обеспечивает прозрачную сериализацию. Слой доступа к данным построен на основе паттерна Repository, инкапсулирующего операции с базой данных и включающего многоуровневую валидацию: проверку уровней компетенций (диапазон 1–3) и проверку существования идентификаторов в таблице `competencies`. Фабрика сессий реализована с использованием контекстного менеджера, обеспечивающего автоматическое управление транзакциями. Настроена система миграций Alembic с автоматическим созданием схемы и механизмом повторных попыток подключения на основе Tenacity.

Разработан backend-сервис на FastAPI, выступающий в роли оркестратора. Приложение построено по модульному принципу с разделением на роутеры аутентификации, пользователей, компетенций, профессий и резюме. Реализована подсистема аутентификации на основе OAuth2 с JWT-токенами и хешированием паролей алгоритмом bcrypt. В целях безопасности сообщения об ошибках аутентификации унифицированы. Реализована двухуровневая модель доступа с разделением прав обычных пользователей и администраторов. Конфигурация вынесена в класс Settings на основе pydantic-settings, чувствительные поля защищены типом SecretStr.

Центральным элементом backend-сервиса является пайплайн анализа резюме в эндпоинте `POST /resumes/analyze`. Разработан класс ResumeParser, поддерживающий форматы PDF (PyPDF2), DOCX (python-docx) и TXT (с обработкой кодировок UTF-8 и CP1251). Взаимодействие с ML-сервисами организовано через `httpx.AsyncClient` с последовательной обработкой: сначала вызывается `ml-extractor` для семантического поиска, затем `ml-grader` для каждой найденной компетенции. Реализована агрегация результатов с вычислением процента соответствия, формированием списка недостающих навыков и рекомендаций по альтернативным профессиям.

Разработаны API-обёртки для ML-микросервисов `ml-extractor` и `ml-grader` в виде независимых FastAPI-приложений. Для обеспечения отказоустойчивости межсервисного взаимодействия реализованы: таймауты HTTP-запросов, механизм повторных попыток с экспоненциальной задержкой (максимум 3 попытки, задержка от 1 до 10 секунд), обработка частичных результатов (продолжение анализа при ошибке оценки одной компетенции) и детальное логирование всех вызовов.

Выполнена контейнеризация всех компонентов системы. Backend-сервис упакован с использованием многоэтапной сборки Docker, минимизирующей размер образа. ML-микросервисы используют отдельные Dockerfile с прямой установкой PyTorch и трансформерных библиотек. Настроена оркестрация через Docker Compose с четырьмя сервисами, healthcheck-проверками для базы данных, пробросом переменных окружения и сетевым взаимодействием. Управление конфигурацией реализовано через переменные окружения в соответствии с методологией 12-Factor App.

В результате выполнения второго раздела получен полностью функциональный программный комплекс, готовый к развёртыванию в производственной среде.

ЗАКЛЮЧЕНИЕ

В рамках данной выпускной квалификационной работы была спроектирована и реализована серверная часть программного комплекса «HR-Интеллект» — интеллектуальной системы автоматизированного анализа резюме и оценки профессиональных компетенций кандидатов. Основной целью работы являлась разработка надёжной, масштабируемой и расширяемой программной инфраструктуры, обеспечивающей функционирование ML-моделей и предоставляющей удобный интерфейс для HR-специалистов.

В ходе выполнения работы были решены следующие задачи. Проведён системный анализ предметной области и выполнено концептуальное моделирование данных, в результате которого выделены ключевые сущности системы и спроектирована ER-диаграмма базы данных. Разработана логическая и физическая модель реляционной базы данных на PostgreSQL, включающая таблицы для хранения пользователей, компетенций, профессий с матрицами требований, ссылок-приглашений, резюме и результатов анализа. Обосновано использование типа JSONB для хранения сложноструктурированных данных, сочетающего гибкость документоориентированного подхода с производительностью реляционной СУБД.

Спроектирована архитектура распределённой системы, основанная на микросервисном подходе. Система декомпозирована на пять компонентов: backend-оркестратор, два ML-микросервиса (ml-extractor и ml-grader), базу данных и клиентское веб-приложение. Разработаны UML-диаграммы компонентов и последовательности, описывающие структуру системы и потоки данных между сервисами. Спроектирована спецификация REST API для всех межсервисных взаимодействий, определены форматы запросов и ответов, коды статусов и механизмы обработки ошибок.

Практическая реализация включала разработку слоя доступа к данным с использованием SQLAlchemy и паттерна Repository, реализацию системы миграций Alembic с автоматическим созданием схемы базы данных и механизмом повторных попыток подключения. Разработан backend-сервис на FastAPI, реализующий аутентификацию с JWT-токенами, ролевою моделью доступа, CRUD-операции для управления справочниками и центральный пайплайн анализа резюме. Реализованы API-обёртки для ML-микросервисов с механизмами таймаутов, повторных запросов и логирования. Выполнена контейнеризация всех

компонентов с использованием Docker и настроена оркестрация сервисов через Docker Compose, что позволяет развернуть весь программный комплекс одной командой.

Разработанное решение обладает рядом преимуществ. Микросервисная архитектура обеспечивает возможность независимого масштабирования компонентов: при росте нагрузки вычислительные ресурсы могут быть добавлены только тому сервису, который является узким местом. Изоляция ML-моделей в отдельных сервисах позволяет обновлять их независимо от основного приложения. Использование JSONB для хранения матриц компетенций обеспечивает гибкость при изменении требований к профессиям без необходимости изменения схемы базы данных. Стандартизированный REST API упрощает интеграцию с внешними системами и разработку альтернативных клиентских приложений.

Важно отметить, что данный проект реализован в рамках программы «Стартап как диплом», что подчёркивает его прикладной характер и потенциал дальнейшего развития в качестве коммерческого продукта.

К возможным направлениям дальнейшего развития системы можно отнести: добавление фоновой обработки резюме с использованием очередей сообщений для повышения пропускной способности; переход к параллельной обработке компетенций в ml-grader для сокращения времени анализа; внедрение кеширования результатов анализа для повторных запросов; расширение API для интеграции с внешними HRM-системами; добавление мониторинга и сбора метрик для оценки производительности ML-сервисов в production-окружении.