

УДК 004.9

РАЗРАБОТКА СИСТЕМЫ МОНИТОРИНГА ДЛЯ РЕКОМЕНДАТЕЛЬНОЙ СИСТЕМЫ ЕДИНОЙ ЦИФРОВОЙ ПЛАТФОРМЫ «РАБОТА В РОССИИ»

Лесков Дмитрий Вячеславович¹, Сычева Маргарита Петровна²

¹Разработчик;

ООО «Клауд Ком»;

Россия, 141983, Московская область, г. Дубна, ул. Программистов, 4;

e-mail: dmtrl.leskov@yandex.ru.

²Разработчик CFM-проектов;

ООО «Аритмос»;

141980, Московская обл., г. Дубна, ул. Дачная 1;

Старший преподаватель;

Государственный университет «Дубна»;

Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;

e-mail: Sytcheva.m.p@uni-dubna.ru.

В данной работе представлена разработка и внедрение системы мониторинга для рекомендательной системы, интегрированной в единую цифровую платформу «Работа в России». Система использует современные методы анализа данных и модели больших языковых моделей (LLM), а также методы А/В тестирования для оптимизации и улучшения функционирования рекомендательных систем. Разработка включает проектирование клиентской и серверной частей приложения, определение функционально-технических требований и проведение тестирования.

Ключевые слова: мониторинг, рекомендательная система, LLM, А/В тестирование, «Работа в России».

Для цитирования:

Лесков Д. В., Сычева М. П. Разработка системы мониторинга для рекомендательной системы единой цифровой платформы «Работа в России» // Системный анализ в науке и образовании: сетевое научное издание. 2024. № 4. С. 1-13. EDN: EXEUF. URL: <https://sanse.ru/index.php/sanse/article/view/631>.

DEVELOPMENT OF A MONITORING SYSTEM FOR THE RECOMMENDATION SYSTEM OF THE UNIFIED DIGITAL PLATFORM "WORK IN RUSSIA"

Leskov Dmitriy V.¹, Sytcheva Margarita P.²

¹Developer;

LLC "Cloud Com";

4 Programmistskaya Str., Dubna, Moscow region, 141983, Russia;

e-mail: dmtrl.leskov@yandex.ru.

²CFM project developer;

LLC "Aritmos";

1 Dachnaya Str., Dubna, Moscow region, 141980, Russia;

Senior teacher;

Dubna State University;

19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;

e-mail: Sytcheva.m.p@uni-dubna.ru.

This paper presents the development and implementation of a monitoring system for a recommendation system integrated into a unified digital platform "Work in Russia". The system uses modern methods of data analysis and large language models (LLM), as well as A/B testing methods to optimize and improve the



Статья находится в открытом доступе и распространяется в соответствии с лицензией Creative Commons «Attribution» («Атрибуция») 4.0 Всемирная (CC BY 4.0) <https://creativecommons.org/licenses/by/4.0/deed.ru>

functioning of recommendation systems. The development includes designing the client and server parts of the application, defining functional and technical requirements and testing.

Keywords: monitoring, recommendation system, LLM, A/B testing, “Work in Russia”.

For citation:

Leskov D. V., Sytcheva M. P. Development of a monitoring system for the recommendation system of the unified digital platform “Work in Russia”. *System analysis in science and education*, 2024;(4):1-13 (in Russ). EDN: EXEUF. Available from: <https://sanse.ru/index.php/sanse/article/view/631>.

Введение

В наше время, чтобы оставаться востребованным специалистом, недостаточно только следить за рынком труда. Просматривая вакансии на профильных сайтах, можно получить исчерпывающую информацию о ситуации на рынке, которая позволит найти стабильную и интересную работу, на которую можно с удовольствием ходить каждый день на протяжении многих лет. Одним из таких профильных сайтов является единая цифровая платформа «Работа в России».

«Работа в России» – это онлайн-платформа, предназначенная для поиска работы и развития карьеры в России. На этом портале можно найти широкий выбор вакансий от различных работодателей, а также получить информацию о требованиях к кандидатам, условиях работы и возможностях для профессионального роста.

Основные возможности портала включают в себя:

1. поиск вакансий по различным критериям, таким как отрасль, регион, должность, уровень зарплаты;
2. создание профиля соискателя, в котором можно указать свой опыт работы, образование, навыки и другую информацию, которая поможет работодателям оценить вашу квалификацию;
3. возможность подписаться на уведомления о новых вакансиях, соответствующих интересам и требованиям соискателя;
4. возможность связаться с работодателями напрямую через портал и отправить резюме на интересующие вакансии;
5. полезные статьи и советы по трудоустройству, развитию карьеры и профессиональному росту.

В рамках разрабатываемой системы осуществляется анализ результатов применения *GPT* с целью улучшения резюме для соискателя и помощи работодателю при выборе кандидатов на онлайн-портале «Работа в России». Под мониторингом подразумевается непрерывный сравнительный анализ работы различных моделей *GPT*.

После загрузки резюме оно автоматически подвергается обработке с использованием модели *GPT*, которая выполняет следующие этапы:

1. определение уровня токсичности текста;
2. выявление ключевых навыков соискателя;
3. краткое изложение содержания и сопоставление резюме с требованиями вакансии для работодателя;
4. общая оценка профиля соискателя.

Под понятием «токсичности» понимается присутствие в тексте резюме нецензурной лексики, грубого обращения или выражений пренебрежительного характера. Для учета версии модели *GPT* и даты создания запроса используется веб-сервис мониторинга, где пользователи имеют возможность оценить каждый ответ, полученный от модели *GPT*, и оставить комментарий, что позволяет дополнительно обучать модель для достижения более точных результатов. Кроме того, в рамках проекта проводится *A/B* тестирование различных конфигураций модели *GPT* с целью оптимизации ее функционирования.

1. Анализ существующих решений

Система мониторинга представляет собой веб-приложение, предназначенное для наблюдения за результатом работы нейросети *LLM*, с возможностью оценивания и комментирования результатов с целью улучшения её работы, а также наблюдение за результатом работы *LLM* с различной конфигурацией через *A/B* тестирование.

В настоящее время существуют иностранные готовые программные решения, которые можно было использовать при реализации системы.

Среди систем мониторинга можно выделить следующие:

1. “*Prometheus*”: оптимизированная система мониторинга для сбора и анализа временных рядов данных;
2. “*Nagios*”: инструмент для мониторинга сетей, систем и инфраструктуры, обеспечивающий оповещения и отчеты о состоянии;
3. “*Zabbix*”: платформа для мониторинга и управления производительностью ИТ-инфраструктуры с широкими возможностями для настройки и автоматизации.

Среди систем для *A/B* тестирования выделяются следующие готовые решения:

1. “*Google Optimize*”: инструмент от *Google* для проведения *A/B* тестов и персонализации веб-сайтов, интегрированный с *Google Analytics*;
2. “*Optimizely*”: платформа для экспериментов и оптимизации, предлагающая *A/B* тестирование и функциональность персонализации;
3. “*VWO*”: комплексное решение для *A/B* тестирования, многовариантного тестирования и оптимизации конверсий, ориентированное на улучшение пользовательского опыта.

Для анализа были выбраны именно эти системы, поскольку они являются лидерами на рынке по версии различных специализированных рейтингов таких как “*Top Network Monitoring Tools*” [1] и “*Top Conversion Rate Optimization Tools*” [2].

Рассмотрим выбранные аналоги и сравним их критерии по балльно-рейтинговой системе по следующим критериям:

хранение данных:

- «0» – данные хранятся на стороне сервиса;
- «0.5» – данные хранятся частично на стороне сервиса и в собственной базе данных;
- «1» – данные хранятся в собственной базе данных.

кастомизация:

- «0» – отсутствие кастомизации;
- «0.5» – частичная кастомизация;
- «1» – полная кастомизация.

экономическая эффективность:

- «0» – постоянные лицензионные платежи;
- «0.5» – снижение затрат в долгосрочной перспективе;
- «1» – отсутствие платежей.

поддержка и развитие:

- «0» – полная зависимость от поставщика сервисов;
- «0.5» – частичная оперативность поддержки;
- «1» – оперативное внесение изменений.

Результаты анализа представлены в таблице 1–2.

Табл. 1. Сравнительный анализ систем мониторинга.

	«Prometheus»	«Nagios»	«Zabbix»
Хранение данных	0	0	0
Кастомизация	0.5	0.5	0.5
Экономическая эффективность	0	0	0
Поддержка и развитие	0	0.5	0

Табл. 2. Сравнительный анализ систем A/B тестирования

	«Google Optimize»	«Optimizely»	«VWO»
Хранение данных	0	0	0
Кастомизация	0.5	0.5	0.5
Экономическая эффективность	0	0.5	0
Поддержка и развитие	0	0.5	0

Из таблиц видно, что существующие популярные системы не соответствуют большинству ключевых критериев. Основные недостатки включают в себя:

- невозможность хранения данных в своей базе;
- ограниченная возможность кастомизации систем;
- слабая поддержка.

В связи с этим принято решение разработать собственные системы мониторинга и A/B тестирования, которые будут удовлетворять всем вышеперечисленным критериям. Будущее решение позволит создать устойчивую и надежную инфраструктуру, полностью соответствующую политике импортозамещения, что является главным требованием Заказчика.

Разрабатываемая система мониторинга функционирует на основе модели *GPT*, представляющей собой глубокую нейронную сеть, обученную на обширных массивах текстовых данных и способную генерировать текст, учитывая предшествующий контекст сообщений.

Первоначальный этап обработки резюме связан с выявлением уровня токсичности текста, включая анализ наличия непристойной лексики, грубого обращения или пренебрежительных выражений. Для этой цели используется алгоритм, основанный на анализе тональности и структуры предложений.

Последующий этап включает выделение ключевых навыков соискателя, что способствует более быстрой оценке, подходит ли данный кандидат для вакансии. Этот этап основывается на анализе ключевых слов и фраз, описывающих опыт и квалификации соискателя

Для оценки эффективности работы системы и оптимизации ее параметров используется метод A/B тестирования. Пользователям предоставляются различные конфигурации моделей *GPT* и *LLM*, и результаты их работы сравниваются. Полученные данные позволяют оптимизировать параметры моделей для достижения наилучших результатов. A/B тестирование представляет собой методологию для оценки изменений в продукте или услуге путем сравнения двух версий (A и B). Пользователи разделяются на две группы: одна группа видит текущую версию (A), в то время как другая группа видит измененную версию (B). Затем анализируются показатели обеих групп с целью определения, какая конфигурация *GPT* и *LLM* дает лучший результат.

2. Требования к системе

Опишем функциональные и требования для подсистемы *LLM* и *GPT*.

Вкладка «A/B тестирование»:

1. пользователь должен иметь возможность просматривать информацию обо всех экспериментах. Создавать свои эксперименты. Добавлять участников в эксперимент и новые этапы эксперимента.

- по умолчанию информация об экспериментах отображается в виде таблицы;
- при нажатии на уникальный идентификатор эксперимента происходит переход на карточку с подробным описанием эксперимента;
- на карточке эксперимента можно полностью управлять выбранным экспериментом.

Вкладка «GPT»:

- пользователь должен иметь возможность просматривать информацию обо всех ответах GPT. Просматривать отдельно информацию о конкретном резюме и ответов для него, а также выставлять оценку каждому ответу. Так же пользователь должен видеть общую статистику по оценкам для ответов GPT;
- пользователь должен иметь возможность просматривать информацию обо всех ответах GPT;
- по умолчанию пользователю предоставляется информация обо всех резюме в виде таблицы. Пользователю должна быть доступна возможность раскрыть каждое резюме и получить информацию по ответам от GPT.

Так же были определены следующие не функциональные требования:

- сбор данных: система должна иметь возможность собирать данные из различных источников, таких как логи, метрики и т. д.
- отображение информации: пользователям должна быть предоставлена информация в удобном и понятном виде, например, в виде графиков, диаграмм, отчетов и т. д.;
- оповещения и уведомления: система должна предоставлять возможность настройки оповещений и уведомлений о проблемах или событиях, требующих внимания;
- хранение данных: должна быть предусмотрена возможность хранения данных для последующего анализа и отслеживания тенденций;
- безопасность: обеспечение безопасности данных и доступа к системе должно быть одним из ключевых аспектов разработки

3. Архитектура системы

На стадии проектирования были подготовлены описания баз данных и их логическая схема. Далее рассмотрены основные таблицы баз данных (см. рис. 1–2)



Рис. 1. Физическая схема базы данных для A/B тестирования



Рис. 2. Физическая схема базы данных для LLM

После обработки результат выгружается и сохраняется в соответствующую базу данных. Если соискатель как-то изменяет резюме (к примеру, обновляет информацию о своих предыдущих обязанностях), то весь процесс обработки начинается заново (см. рис. 3).

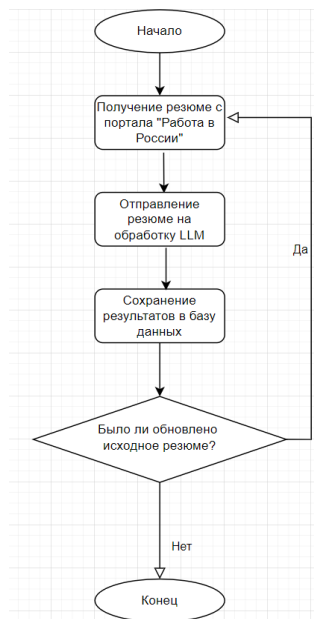


Рис. 3. Алгоритм применения LLM для обработки резюме

Поскольку информация о резюме и результаты работы *LLM* хранятся в отдельных базах данных, требуется разработать эффективный алгоритм, который бы обеспечивал получение резюме и его обработку с целью отображения данных на веб-сервисе (см. рис. 4).

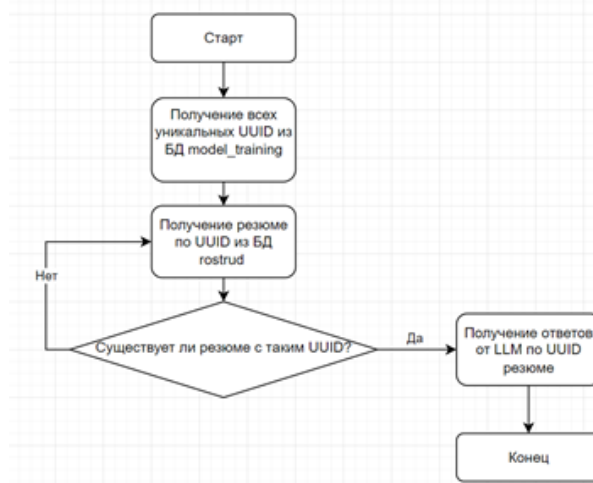


Рис. 4 Алгоритм получения резюме и ответов от GPT

На рис. 5 изображена диаграмма классов сервиса A/B тестирования.

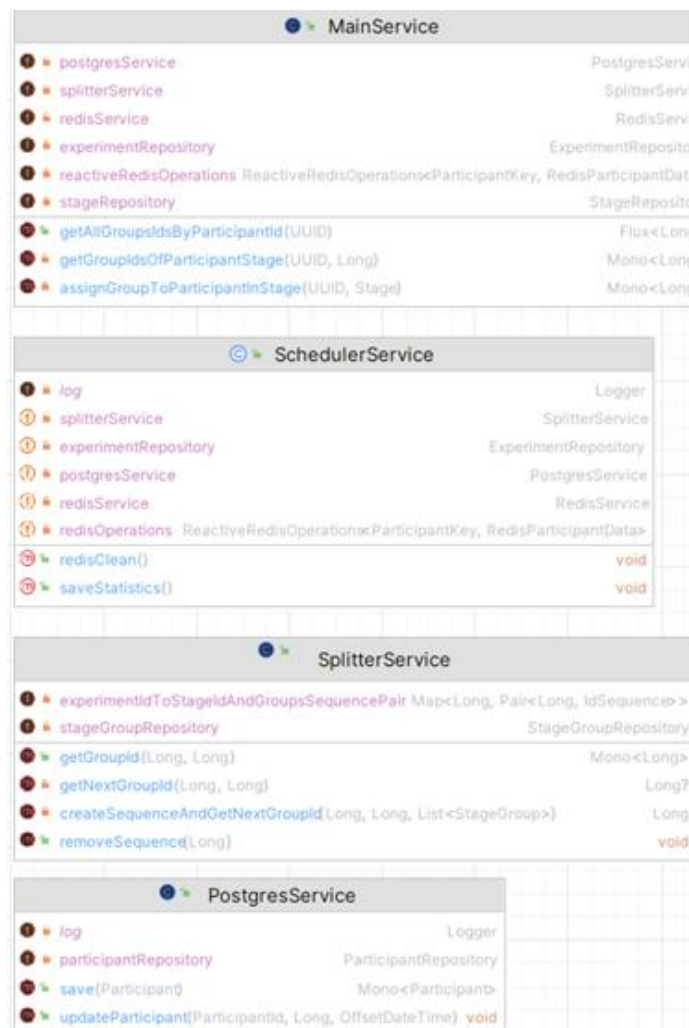


Рис. 5. Диаграмма классов. Сервис A/B тестирования

На рис. 6 изображена диаграмма классов для оценивания работы применения *LLM*.

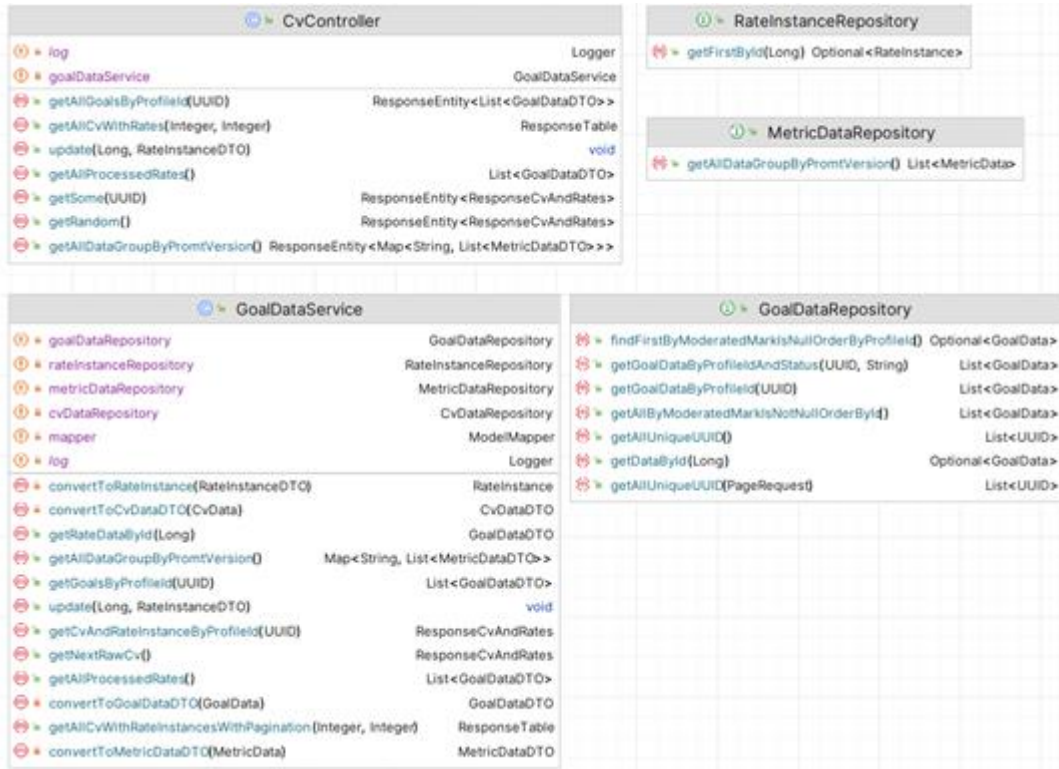


Рис. 6. Диаграмма классов. Сервис оценивания работы LLM

4. Проектирование интерфейса

В работе были спроектированы вкладки для A/B тестирования и оценивания работы LLM. На всех рисунках представлены тестовые данные для защиты личной информации пользователей.

На вкладке со всеми резюме (см. рис. 7) представлена таблица со всеми резюме, прошедшими через LLM. В таблице представлена основная информация о резюме, а также дополнительная таблица для каждого резюме, содержащая последние ответы от LLM.

ID	Описание	Навыки
14398019	Работник торгового зала, контролер	быстро обучаюсь, стрессоустойчивость, внимательность, пунктуальность.

ID	Цель	Версия модели	Оценка	Комментарий	Версия модели
20611	Оценка профиля. Указание моментов которые можно улучшить	07.03.2024	☆☆☆☆☆		07.03.2024
20543	Список навыка соискателя	07.03.2024	☆☆☆☆☆		07.03.2024
20541	Краткое описание профиля	07.03.2024	☆☆☆☆☆		07.03.2024
20542	Определение токичности текста	07.03.2024	☆☆☆☆☆		07.03.2024

Рис. 7. Вкладки с таблицей резюме

На рис. 8 изображена карточка резюме, которая открывается при нажатии на уникальный идентификатор резюме.

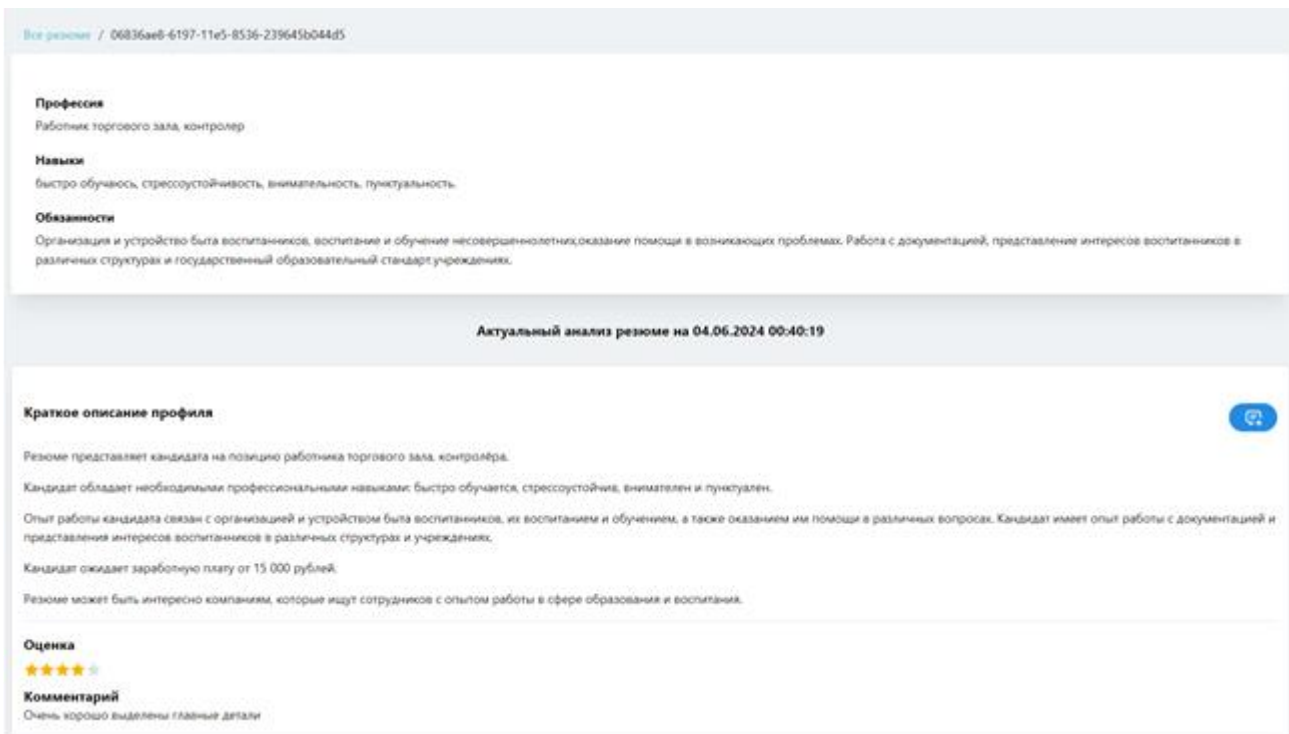
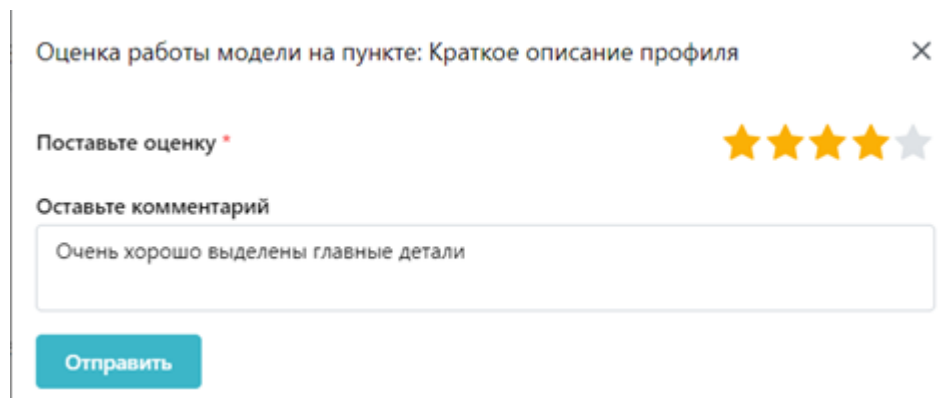


Рис. 8. Карточка резюме

Сверху на карточке находится полное описание резюме, а снизу полные ответы от *LLM*. При нажатии на кнопку справа от названия критерия, по которому проверялось резюме, откроется модальное окно для выставления оценки ответа и оставления комментария в случае необходимости (см. рис. 9)

Рис. 9. Модальное окно оценивания ответа *LLM*

На рисунке 10 изображена таблица, содержащая основную информацию об экспериментах *A/B* тестирования

ИД Т1	Время начала эксперимента Т1	Время окончания эксперимента Т1	Описание Т1	Действия
82	24.04.2024 14:07:28	24.04.2024 14:08:36	Тест	Эксперимент окончен
83	24.04.2024 14:49:15	Эксперимент не закончен	Тест	Завершить эксперимент
85	Эксперимент не начал	Эксперимент не закончен	sa8asd	Запустить эксперимент

Рис. 10. Таблица всех экспериментов *A/B* тестирования

При нажатии на кнопку «Создать эксперимент» откроется модальное окно с формой, при заполнении которого можно создать новый эксперимент (см. рис 11).

Создание эксперимента

Описание
введите описание

Выберите сервисы
выбрать сервис

Распределение групп

Введите процент распределения групп
0.00

Название
введите название группы

Введите процент распределения групп
Значение должно быть от 0 до 1

Название
введите название группы

Добавить группу

Значение контрольной группы
0.00

Название
введите название группы

Создать эксперимент

Рис. 11. Форма создания нового эксперимента

При нажатии на уникальный идентификатор откроется карточка эксперимента (см. рис. 12), содержащая всю информацию о проводимом эксперименте, такую как:

- участники эксперимента;
- этапы эксперимента;
- распределение групп внутри эксперимента.

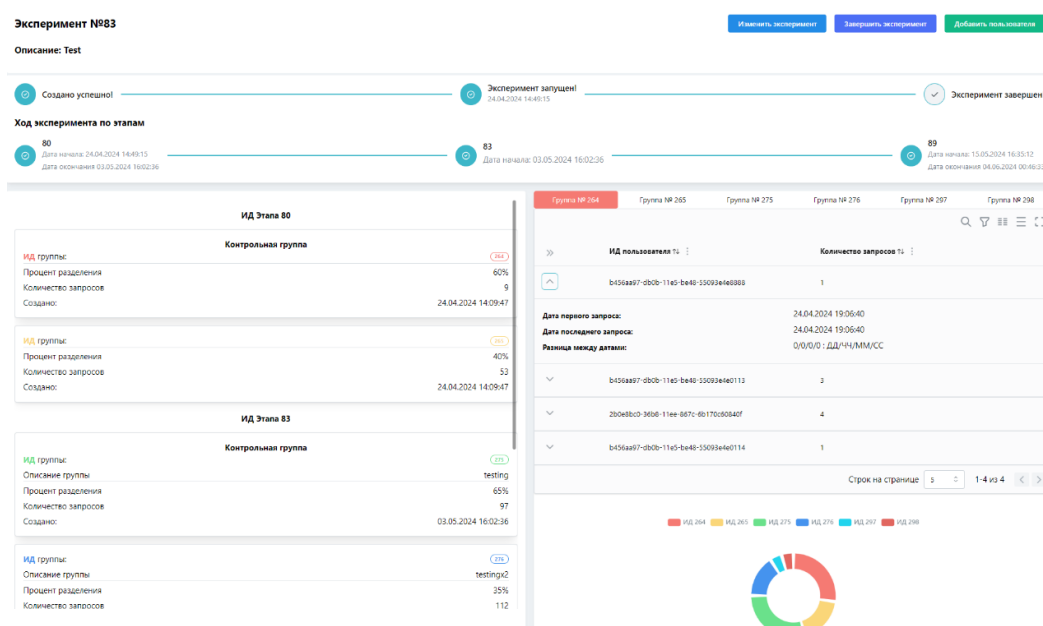


Рис. 12. Карточка эксперимента

5. Технологии реализации

Для реализации описанного решения были использованы следующие программные средства:

1. *Java* – это объектно-ориентированный язык программирования общего назначения, который отличается строгой типизацией. Он был разработан компанией *Sun Microsystems* (в настоящее время принадлежит *Oracle*) и поддерживается сообществом через *Java Community Process*.
2. *JavaScript* – язык программирования, который поддерживает несколько парадигм: объектно-ориентированный, императивный и функциональный.
3. *IntelliJ IDEA* – это интегрированная среда разработки программного обеспечения, созданная компанией *JetBrains*, которая поддерживает множество языков программирования, таких как *Java*, *JavaScript*, *Python*.
4. *WebStorm* – это интегрированная среда разработки (*IDE*) для веб-разработки, разработанная компанией *JetBrains*. Она предоставляет разработчикам удобное окружение для создания веб-приложений с использованием *HTML*, *CSS*, *JavaScript* и других технологий.
5. *Spring Boot* – это популярный фреймворк для создания веб-приложений с использованием *Java*. Это часть экосистемы *Spring*, которая представляет собой набор инструментов и библиотек для создания приложений корпоративного уровня. Он был выбран, потому что позволяет легко и быстро разрабатывать веб-приложения.
6. *Hibernate* – это мощный фреймворк для языка программирования *Java*, который обеспечивает удобный способ работы с базами данных через объектно-ориентированный подход.
7. *PostgreSQL* – это мощная объектно-реляционная система управления базами данных (СУБД), которая предоставляет расширенные возможности для хранения и обработки данных.
8. *React* – это библиотека на *JavaScript*, разработанная *Facebook*, предназначенная для построения пользовательских интерфейсов веб-приложений.
9. *React Query* – это библиотека для управления состоянием и кэширования данных в *React* приложениях.
10. *HTML* – это язык разметки гипертекста, который используется для создания веб-страниц.
11. *CSS* – это формальный язык, который используется для стилизации и описания внешнего вида документа, такого как веб-страница.
12. *Maven* – это инструмент для управления проектами и сборки программного обеспечения на *Java*.
13. *Vite* – это инструмент, который предназначен для быстрой сборки современных веб-приложений.

6. Внедрение и тестирование

По окончании этапа активной разработки программного продукта он был передан в ручное тестирование внутри компании для тщательной проверки всех функций и возможностей приложения. После успешного завершения внутреннего этапа ручного тестирования приложение будет передано заказчику для проведения второго этапа тестирования, осуществляемого на стороне заказчика.

Заключение

В рамках данной работы для достижения поставленной цели был выполнен ряд следующих задач:

1. изучены функционально-технические требования;
2. обозначены возможности проектируемого решения;
3. определены технологии реализации; спроектировано и реализовано, подготовлено к внедрению программное решение.

Программное обеспечение включает в себя следующие ключевые компоненты:

1. система мониторинга сервисов;
2. веб-интерфейс мониторинга сервисов;
3. модуль уведомлений и оповещений.

Основная цель системы – обеспечить удобный и эффективный мониторинг сервисов, входящих в состав портала «Работа в России».

На данный момент программное обеспечение проходит завершающие этапы тестирования внутри команды тестировщиков.

Стоит отметить, что реализованные на данный момент сервисы мониторинга не являются исчерпывающими для всей системы. Для обеспечения полного охвата мониторинга необходимо добавить дополнительные модули и разделы. В связи с этим работа по развитию системы продолжается, с целью дальнейшего расширения ее функциональности и повышения надежности.

Список источников

1. Top Network Monitoring Tools // Optimization | Heatmaps, Recordings, Surveys & A/B Testing — 2022. — URL: <https://www.crazyegg.com/> (дата обращения: 05.04.2024).
2. Морозкин, Н. А. О важности АВ-тестирования при разработке систем Искусственного Интеллекта / Н. А. Морозкин, И. Д. Ромайкин // Инновации. Наука. Образование. – 2021. – № 47. – С. 1694-1699.
3. Список лучших систем А/В тестирования — [Электронный ресурс]. — Электрон. текст. — 2019. — URL : <https://neilpatel.com/de/blog/wie-man-einen-ab-split-test/> (дата обращения: 05.04.2024).
4. Смирнова, Е. А. Удаленная занятость: анализ вакансий и резюме // Лидерство и менеджмент. – 2022. – Т. 9, № 1. – С. 161-172. – DOI: 10.18334/lim.9.1.114298.
5. Ящук, О.О. Последовательный анализ в задачах А/В – тестирования / О.О. Ящук, Я.С. Бондаренко // Вопросы прикладной математики и математического моделирования. — Днепропетровск : РВВ ДНУ, 2016. — URL: <https://pm-mm.dp.ua/index.php/pmmm/article/view/185> (дата обращения: 02.03.2024).
6. Преображенский, Ю. П. Анализ подходов к формированию рекомендательных систем / Ю. П. Преображенский, В. М. Коновалов // Вестник Воронежского института высоких технологий. – 2019. – № 4(31). – С. 88-90.
7. Комарова, Л. А. Сравнение языковых моделей в задаче извлечения навыков из вакансий и резюме / Л. А. Комарова, В. И. Соловьев, А. М. Колосов // Современные информационные технологии и ИТ-образование. – 2024. – Т. 20, № 2. – С. 251-261.