

0.1. Гридяев Е.А., Немоляев Т.В., Казакова Д.И., Седойкин Р.С., Альгашев Г.А. Система автоматизированного мониторинга рейтингов абитуриентов в конкурсных списках

В настоящей работе рассматривается разработка автоматизированной информационной системы мониторинга конкурсных списков абитуриентов, направленной на обеспечение оперативного учёта динамики приёмной кампании и повышение прозрачности процедур формирования рейтинговых списков. Актуальность обоснована увеличением числа поступающих и усложнением правил приёма, включая последовательную перестановку приоритетов, возврат оригиналов документов и массовые обновления заявлений, что исключает адекватное применение ручных и полуавтоматических средств обработки данных и требует формализованных процедур аудита и автоматического пересчёта рейтингов [1].

Цель проекта заключается в проектировании модульной и масштабируемой системы, обеспечивающей импорт входных данных, их формальную валидацию по заданным правилам, корректное формирование конкурсных списков с учётом приоритетов заявлений и проходных баллов, а также сохранение полной истории изменений для целей проверки соответствия регламента и воспроизводимости решений. Для достижения этой цели предложена клиент-серверная архитектура с выделением функциональных подсистем обработки входных данных, программной реализации алгоритмов формирования и пересчёта списков, подсистемы хранения журналов операций с временными метками и идентификацией инициаторов, а также компонент пользовательского интерфейса для оперативной работы сотрудников приёмной комиссии. Данные логически структурированы вокруг трёх основных сущностей: «абитуриент» (оценки, набор приоритетов), «направление подготовки» (число вакантных мест, историческая статистика) и «запись журнала операций» (временная метка, тип операции, инициатор), что упрощает формализацию бизнес-правил и расширение предметной модели.

В качестве технической реализации предпочтена комбинация технологий, обеспечивающая детерминированность поведения и возможности масштабирования: серверная логика реализована с использованием платформы Spring Boot (проект на языке Java), компонент пользовательского интерфейса — с применением библиотеки React.js [2] для построения интерактивных представлений, для визуализации метрик используются специализированные библиотеки визуализации данных [3], а в качестве системы управления базами данных рекомендуется PostgreSQL; процессы развёртывания и обеспечения качества реализуются посредством контейнеризации и механизмов непрерывной интеграции и непре-

рывного развёртывания. Такая реализация обеспечивает разделение ответственности между слоями системы, упрощает модульное тестирование и даёт возможность горизонтального масштабирования при росте нагрузки. Важным компонентом архитектуры является подсистема журналирования: каждая транзакция и изменение записываются с указанием времени, типа операции и идентификатора пользователя, что гарантирует воспроизводимость состояния и поддержку процедур внутреннего и внешнего контроля.

Интерфейсные решения предусматривают представление рейтингов в табличном виде с многоуровневой фильтрацией и возможностями сортировки, визуализацию динамики подачи и отзыва документов и агрегированные показатели конкуренции по направлениям подготовки для принятия управленческих решений. Верификационные испытания в имитированной среде, моделирующей нагрузку крупного высшего учебного заведения, показали приемлемое время обработки: массовая загрузка и валидация десятков тысяч записей выполняются за несколько секунд, а критические операции пересчёта рейтингов выполняются в доли секунды, что обеспечивает предоставление практически актуальной информации сотрудникам приёмной комиссии.

Сравнительный анализ с традиционными подходами (например, электронными таблицами) и универсальными информационными системами демонстрирует преимущества предложенного решения в части формализации бизнес-правил, обеспечения полноты журнала операций и расширенных средств аналитики и визуализации; модульная архитектура обеспечивает адаптивность к изменениям нормативных требований и облегчает интеграцию дополнительных функциональных модулей [4]. Практическая значимость заключается в уменьшении объёма ручной работы, снижении числа ошибок при формировании конкурсных списков и повышении прозрачности процедур приёма. Для дальнейшего развития рекомендуется направление исследований и разработок в сторону внедрения методов прогнозной аналитики на основе машинного обучения для аппроксимации проходных баллов, расширения интеграции с внешними информационными системами и оптимизации алгоритмов распределения с учётом сложных правил приёма, что позволит повысить точность прогнозов и адаптивность системы к изменяющимся требованиям приёмных кампаний.

Список литературы

- [1] Божович М. Поступление похоже на игру в покер. Эксперт — о проекте новых правил приёма в вузы-2025 [Электронный ресурс]. URL: <https://www.pravmir.ru/postuplenie-pohozhe-na-igruv-poker-ekspert-o-proekte-novyh-pravil-priema-v-vuzy-2025/> (дата обращения: 31.07.2025).

- [2] Суздальницкий И. React.js: понятное руководство [Электронный ресурс]. URL: <https://habr.com/ru/companies/ruvds/articles/428077/> (дата обращения: 31.07.2025).
- [3] OLUWADAMISI S. How to use Chart.js for interactive data visualization [Электронный ресурс]. URL: <https://www.freecodecamp.org/news/how-to-use-chart-js-for-interactive-data-visualization/> (дата обращения: 31.07.2025).
- [4] ГЕНЕРАЛОВ И. Г., АЛЕКСЕЕВА Л. А. Место «1С: Университет» среди информационных технологий [Электронный ресурс]. URL: <https://cyberleninka.ru/article/n/mesto-1s-universitet-sredi-informatsionnyh-tehnologiy/viewer> (дата обращения: 31.07.2025).