Некоммерческое акционерное общество  
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ  
имени Гумарбека Даукеева»  
Институт автоматизации и информационных технологий  
Кафедра IT-инженерии и искусственного интеллекта

**ОТЧЕТ**

**по проекту**

По дисциплине: Разработка сетевых приложений:

На тему: Инструмент для проведения опросов и анкетирования среди исследователей

Образовательная программа 6B06102 – Информационные системы

Выполнил: Котов Д.А. , Дюсембаев Ж.К. , Жербаков А.Е.   
 Группа: ИСу-23-1

Проверил: Старший преподаватель АЛДИЯРОВ М.А.

\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г.  
(оценка) (подпись)

Алматы 2025

**1.ВВЕДЕНИЕ**

Постановка задачи

Научные исследования, социальные опросы, маркетинговый анализ и другие виды исследований требуют проведения точного сбора данных. Анкетирование и опросы остаются одними из наиболее востребованных методов, позволяя получать важную информацию о мнениях, предпочтениях и потребностях респондентов. Однако современные исследователи сталкиваются с рядом проблем: сложностью создания анкет, громоздкостью инструментов для обработки результатов и необходимостью интеграции данных в аналитические системы.

Разработка веб-инструмента для проведения опросов должна удовлетворять следующим ключевым требованиям:

Удобство: Простой и интуитивно понятный интерфейс для создания и настройки опросов.

Автоматизация: Минимизация ручного труда, автоматический сбор и первичная обработка данных.

Доступность: Обеспечение работы с любыми устройствами и возможность работы в удалённом режиме.

Аналитические функции: Наличие инструментов для анализа данных, включая визуализацию и генерацию отчетов.

Проект направлен на решение этих задач путем создания удобного и функционального инструмента, отвечающего запросам исследовательской аудитории.

Актуальность темы

Собирая данные для исследований, исследователи вынуждены использовать либо устаревшие методы, такие как бумажные анкеты, либо сложные многофункциональные системы, требующие значительного времени для обучения. Это тормозит процесс анализа данных и приводит к недостаточному использованию технологий автоматизации.

На фоне глобальной цифровизации проблема автоматизации научных исследований приобретает особую актуальность.

Возрастающая нагрузка на исследователей. С увеличением объема данных ручной сбор и обработка становятся непродуктивными.

Технический разрыв. Большинство доступных инструментов либо слишком просты (например, Google Forms), либо перегружены функциями, которые не востребованы обычными пользователями.

Потребность в адаптивности. Исследователи всё чаще требуют возможности настройки инструментов под свои задачи, включая поддержку уникальных форматов данных.

Таким образом, разработка инструмента, который упростит проведение опросов и анализ результатов, отвечает современным вызовам и имеет прикладное значение для науки и бизнеса.

Цель и задачи проекта

Целью проекта является разработка универсального веб-инструмента для проведения опросов и анкетирования, позволяющего исследователям легко создавать анкеты, собирать данные и анализировать результаты, с учётом потребностей в автоматизации процессов.

Для достижения поставленной цели требуется решить следующие задачи:

1. Провести анализ существующих решений в области опросов и анкетирования.
2. Разработать функционал для создания анкет, гибко настраиваемых под потребности исследователей.
3. Внедрить возможности сбора данных в режиме реального времени.
4. Реализовать аналитический модуль для обработки данных и визуализации результатов.
5. Создать систему автоматической генерации отчетов по результатам опросов.
6. Предусмотреть поддержку интеграции с популярными аналитическими платформами (например, Tableau, Power BI).
7. Обеспечить безопасность данных, включая защиту персональной информации респондентов.

Краткий обзор литературы

Современные исследования в области автоматизации анкетирования и опросов подчеркивают ключевую роль цифровых технологий в повышении эффективности сбора и обработки данных. Значительное внимание уделяется удобству пользовательского интерфейса и адаптивности платформ, поскольку сложные и громоздкие системы могут затруднять работу исследователей и респондентов.

В научных работах и обзорах отмечается, что эффективные инструменты для опросов должны поддерживать персонализацию анкет, гибкую настройку логики вопросов, а также интеграцию с аналитическими системами. Например, использование технологий искусственного интеллекта для анализа ответов позволяет не только автоматизировать обработку данных, но и получать более глубокие аналитические инсайты.

Такие платформы, как Google Forms, SurveyMonkey и Typeform, обеспечивают базовый функционал для создания и управления опросами. Однако их ограничения включают высокую стоимость расширенных функций, недостаточную гибкость в настройке сложной логики анкет и ограниченные возможности интеграции с внешними аналитическими инструментами.

**2. Основные цели:**

Обеспечить возможность регистрации и авторизации пользователей (роли: администратор, участник).

Реализовать функционал создания, редактирования и удаления анкет и вопросов.

Предоставить пользователям интерфейс для прохождения опросов и отправки ответов.

Реализовать отображение результатов опросов в виде графиков и статистики.

Задачи:

Спроектировать и разработать REST API для управления анкетами.

Разработать адаптивный и современный frontend с использованием Vue.js и Bootstrap.

Обеспечить безопасность данных (аутентификация, авторизация, защита от CSRF, XSS).

Организовать хранение данных в реляционной базе данных (PostgreSQL или MySQL).

**3. Анализ требований**

Функциональные требования

Пользовательская часть:

* Регистрация, вход и восстановление пароля.
* Просмотр списка доступных опросов.
* Прохождение опросов: выбор ответов, переход между вопросами.
* Просмотр результатов (индивидуальных и агрегированных статистик) по завершённому опросу.

Административная панель:

* Создание, редактирование, удаление анкет.
* Управление вопросами (текст вопроса, варианты ответов, тип вопроса — одиночный выбор, множественный выбор, текстовый ввод и т.д.).
* Просмотр и анализ статистики по каждому опросу (отчёты, графики).

API:

* CRUD-операции для анкет, вопросов, ответов.
* Эндпоинты для аутентификации (JWT или сессии).
* Документация API (например, с использованием Swagger/OpenAPI).

Нефункциональные требования

* Безопасность: шифрование паролей, защита от CSRF/XSS, валидация входных данных.
* Производительность: быстрая обработка запросов, оптимизация запросов к БД.
* Масштабируемость: возможность добавления новых модулей и интеграция с внешними сервисами.
* Удобство использования: адаптивный дизайн, отзывчивый интерфейс на всех устройствах.
* Кроссбраузерность: поддержка современных браузеров

**4. Технологический стек и инструменты**

Backend:

* Язык: Java
* Фреймворк: Spring Boot (с модулями Spring MVC, Spring Security, Spring Data JPA)
* Сборка: Maven или Gradle
* База данных: PostgreSQL или MySQL

Frontend:

* Фреймворк: Vue.js
* Язык: JavaScript (или TypeScript при необходимости)
* Разметка: HTML5
* Стилизация: Bootstrap (CSS/JS)

Прочие инструменты:

* Система контроля версий: Git (GitHub, GitLab или Bitbucket)
* API-тестирование: Postman
* Интеграционное тестирование: JUnit, Mockito для backend; Vue Test Utils для frontend
* CI/CD: GitHub Actions, Jenkins или аналогичные системы
* Документация API: Swagger/OpenAPI

**5. Архитектура системы**

Слой представления (Frontend):

* SPA на Vue.js.
* Использование HTML и Bootstrap для создания адаптивного и удобного интерфейса.
* Коммуникация с backend через REST API.

Слой бизнес-логики (Backend):

* Spring Boot приложение, реализующее REST-контроллеры.
* Обработка бизнес-логики опросов, пользователей, ответов.
* Использование Spring Security для защиты эндпоинтов.
* Интеграция с базой данных через Spring Data JPA.

Слой данных:

* Реляционная база данных для хранения информации о пользователях, анкетах, вопросах и ответах.
* Возможная организация кеширования для ускорения работы (например, с использованием Redis).

Взаимодействие слоёв:

* Frontend отправляет AJAX-запросы к REST API.
* Backend обрабатывает запросы, обращается к базе данных и возвращает данные в формате JSON.