

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Уральский федеральный университет  
имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РТФ  
Школа бакалавриата

## ОТЧЕТ

По проекту  
«Survey»

по дисциплине «Проектный практикум»

Заказчик: 66bit

Куратор: Путинцева Татьяна Александровна

Студенты команды CrackStatus

Николаева Татьяна Александровна

Русинов Матвей Александрович

Черников Георгий Павлович

Шептихин Вячеслав Алексеевич

---

---

---

---

---

---

---

Екатеринбург, 2025

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1 Описание .....	4
2 Актуальность для заказчика.....	5
3 Требуемый и ожидаемый результат .....	6
4 Основная часть .....	7
4.1 Цель .....	7
4.2 Инструменты для реализации: .....	7
4.3 План работы.....	7
4.3.1 Аналитика .....	8
4.3.2 Рассмотрение конкурентов .....	8
4.3.3 Core .....	12
4.3.4 Веб-интерфейс.....	12
4.4 Этапы разработки.....	14
4.4.1 Первый контрольный пункт.....	15
4.4.2 Второй контрольный пункт .....	15
4.4.3 Третий контрольный пункт.....	15
ЗАКЛЮЧЕНИЕ .....	17
ПРИЛОЖЕНИЕ А (обязательное) Первый контрольный пункт.....	18
ПРИЛОЖЕНИЕ В (обязательное) Второй контрольный пункт.....	20
ПРИЛОЖЕНИЕ С (обязательное) Третий контрольный пункт .....	22

## **ВВЕДЕНИЕ**

## **1 Описание**

Проект направлен на создание API приложения с веб-интерфейсом, которое позволяет легко и быстро создавать опросы и собирать результаты в удобной форме. Приложение должно быть полезно для студентов и преподавателей, которые хотят проводить опросы для сбора мнений или обратной связи по различным вопросам. Отдельно развернутое API позволит в дальнейшем внедрять систему опросов в любые другие проекты.

Пользователи имеют возможность создавать опросы с разными типами вопросов, отправлять ссылки на опросы участникам и просматривать результаты в реальном времени. Также предусмотрена возможность скачивания результатов опросов в формате PDF или Excel для дальнейшего анализа или презентации. Приложение должно быть простым в использовании, с интуитивно понятным интерфейсом, что позволит быстро создавать и управлять опросами.

Необходимо отдельно реализовать работу API и WEB части для дальнейшего внедрения сервиса. Веб-интерфейс необходим для проверки функционала сервиса.

## **2 Актуальность для заказчика**

Компания занимается заказной разработкой. Зачастую заказчики в своей будущей системе хотят видеть функционал опросов. Одно из условий — опросы не должны быть внешней системой, то есть использовать апи популярных систем для опросов нельзя (Яндекс, Гугл и т.д.). Библиотека позволит быстро и удобно внедрять функционал опросов в любую систему, что упростит дальнейшую разработку.

### **3 Требуемый и ожидаемый результат**

На этапе согласования проекта с заказчиком были выделены следующие требования к результату продукта:

- библиотека и/или API: функционал подготовлен для интеграции в другие проекты;
- реализация функционала для создания опросов: возможность добавлять различные типы вопросов, задавать параметры видимости и сроков проведения;
- создание системы управления опросами: возможность редактировать активные опросы, просматривать результаты в реальном времени;
- внедрение функции отправки опросов: генерация уникальных ссылок и возможность отправки опросов через различные каналы (например, по электронной почте);
- реализация функции скачивания результатов: экспорт результатов опросов в формате либо PDF, либо Excel для дальнейшего анализа и сохранения;
- разработка дизайна приложения: создание прототипа интерфейса с учетом удобства использования и интуитивной навигации.

По завершении проекта ожидается разработать полностью исправное и функционирующее приложение, отвечающее требованиям заказчика. После представления проекта заказчику ожидается одобрение разработанного продукта, а также, при необходимости поддержки и доработки проекта, продолжить сотрудничать и разрабатывать актуальные функции.

## **4 Основная часть**

### **4.1 Цель**

Создать удобный инструмент для создания и проведения опросов, который поможет пользователям собирать и анализировать данные, а также сохранять результаты в формате Excel. Весь функционал должен быть вынесен в сервисы и легко интегрируемый в другие проекты.

### **4.2 Инструменты для реализации:**

Заказчиком были выделены следующие ограничения технических требований:

Backend: C# ASP.NET Core, EF Core

Frontend: JavaScript, React, TypeScript

### **4.3 План работы**

В первую очередь необходимо провести аналитику: а именно выявить необходимый функционал для web приложения для демонстрации работы core библиотеки, рассмотреть потенциальные трудности во время процесса разработки, а также проанализировать потенциальных конкурентов. За данный процесс ответственен Русинов М. А.

После представленной аналитики, предоставляется возможность создание технического задания с use case. После согласования технического задания с командой и заказчиком, открывается возможность начать разработку продукта. Во время которой необходимо менеджить задачи в команде, а именно: распределять таски между участниками команды, следить за процессом и оценивать его. Для менеджмента используется YouTrack. За данный этап ответственен Черников Г. П.

Этап разработки включает в себя три основных процесса: составление макетов web-приложения в Figma, разработка core библиотеки, разработка Web-приложения со своим сервером. Этот этап является самым длительным и трудозатратным. В процесс этого этапа задействована вся команда. За дизайн ответственна Сабирьянова В. А. (отчислилась в последствии разработки). Разработку core библиотеки и backend сервера ответственен Шептихин В. А. За разработку web-приложения ответственна Николаева Т. А.

#### **4.3.1 Аналитика**

Необходимо создать библиотеку для удобного взаимодействия с базой данных опросов, а также web-сервис с простым дизайном для демонстрации работы самой библиотеки. Важно поддерживать различные типы вопросов: текстовые, с несколькими вариантами ответа, шкала, дата, и, возможно, вопросы с изображениями. Это позволит обеспечить гибкость для разных сценариев использования. Реализовать функциональность, где администратор может настроить временные ограничения на доступ, а также ограничить доступ на основе авторизации пользователей.

Нужно реализовать генерацию уникальных ссылок на основе токенов или UUID и предоставить механизм рассылки через API или интеграцию с внешними каналами (email, мессенджеры).

Реализовать экспорт данных в .xlsx формате с возможностью настройки шаблонов для отчетов и оформления.

#### **4.3.2 Рассмотрение конкурентов**

Google Forms — бесплатный инструмент для создания опросов с базовой аналитикой, в качестве отдельного сервиса или API.

Плюсы: интуитивно понятный интерфейс, гибкость типов вопросов – текстовые, множественный выбор, шкалы, загрузка файлов и так далее,



автоматическое построение диаграмм и сводных данных, их API легко внедряется и реализовано удобное извлечение данных.

Минусы: нет возможности реализовать кастомный дизайн, в частности потому что нет отдельной библиотеки, сервис привязан к Google аккаунтам, нет открытой возможности интегрировать внешние сервисы, для API нет возможности автоматически генерировать уникальные ссылки, нельзя выгружать данные напрямую в .xlsx формат.

Итог: есть возможность сделать более гибкую и как минимум интегрируемую систему с любыми вариациями дизайна на усмотрение разработчиков, кастомной авторизации и более простой выгрузкой в формат .xlsx.

Yandex Forms — бесплатный сервис для создания простых форм и опросов, интегрированный в экосистему Яндекса.

Плюсы: очень понятный интерфейс, легко создавать формы без технических знаний, основной функционал доступен без оплаты, поддержка текстовых ответов, выбора одного или нескольких вариантов, шкал, загрузки файлов и прочее, имеется возможность получить ссылку или QR-код на форму.

Минусы: официального API для интеграции в сторонние системы нет, нет готовых решений для интеграции в мобильные/веб-приложения как библиотеки, базовая статистика ответов, нет продвинутой сегментации или детальной визуализации данных, возможна выгрузка в CSV, но нет встроенной поддержки экспорта в формате .xlsx, нет полноценной email-рассылки.

Итог: есть возможность реализовать интегрируемую систему, которая не будет мешать разработчикам использовать свои варианты дизайна форм или опросов, с более комплексной системой рассылок и возможностью выгрузки в формате .xlsx напрямую.

SurveyMonkey — мощная платформа для проведения опросов с продвинутой аналитикой, в качестве отдельного сервиса или API.

Плюсы: мощные инструменты анализа данных, включая сегментацию и автоматические отчеты, поддержка различных форматов вопросов, включая условную логику, отправка опросов по email, SMS и соцсетям, кастомизация внешнего вида форм при платной подписке.

Минусы: достаточно ограниченный функционал по множеству опций для бесплатного тарифа, нельзя развернуть локально (not self-hosted), в API нет нативной выгрузки в формате .xlsx, очень узкая кастомизация, особенно для бесплатного тарифа.

Итог: есть возможность предложить более доступное интегрирование опросов в сервис с прямой выгрузкой в формате .xlsx и кастомизацией самих форм/опросов.

SurveyJS — мощная библиотека для создания динамических опросов, которая поддерживает разные платформы и языки программирования, что ближе к идеи проекта.

Плюсы: полноценная библиотека для работы с опросами — можно программно создавать, редактировать и обрабатывать ответы, можно менять внешний вид через CSS или использовать разные рендеринг-движки, поддержка нескольких языков программирования, поддержка экспорта в .pdf, json, .xlsx, есть поддержка Webhooks — можно реагировать на новые ответы в реальном времени.

Минусы: необходимо реализовать серверную часть и настроить базу данных самостоятельно с использованием SurveyJS, так как это библиотека, мешает ограниченная функциональность для бесплатного тарифа, что отрезает половину функциональности, описанной выше.

Итог: очень сильный конкурент, есть возможность предложить более доступное в цене решение в виде библиотеки для более узконаправленных проектов.

Typereform — это облачный сервис для создания опросов и форм с упором на интерактивный пользовательский опыт и гибкость дизайна, но с ограниченными возможностями API и кастомизации backend-логики.

Плюсы: интуитивно понятный и современный сервис с гибкостью дизайна, делающий процесс заполнения опроса более интерактивным и удобным для респондентов, гибкая настройка логики переходов и условий, что позволяет создавать «разговорные» опросы, где вопросы подстраиваются под ответы пользователей, хорошая интеграция с внешними сервисами (CRM, аналитика, email-рассылки) через готовые коннекторы и API.

Минусы: расширенные возможности доступны лишь в платных тарифах, что может быть ограничением для малого бизнеса или стартапов, экспорт данных может требовать дополнительных усилий для кастомизации шаблонов отчётов, особенно если требуется формат .xlsx с уникальным оформлением.

Итог: Typereform предлагает современный, ориентированный на пользователя интерфейс и гибкость в создании опросов, но может оказаться менее доступным с точки зрения цены и кастомизации экспорта данных.

Выделим некоторые преимущества разрабатываемого продукта, которые могут отличать от конкурентов: в первую очередь планируется предложить простую интегрируемость в серверную часть сервисов и доступность всего функционала без платных тарифов. Стоит выделить и несколько других достоинств: прямую выгрузку в формате .xlsx, удобный механизм email-рассылки и кастомизация опросов или форм в плане дизайна, что отчасти следует из интегрируемости сервиса.

В таблице 1, кратко представлены конкуренты по всем критериям.

Таблица 1 – Конкуренты

	Интегрируемость	Доступность	Экспорт в .xlsx	Email-рассылка	Кастомизация
Google Forms	-	+	-	-	-
Yandex Forms	-	+	-	-	-
SurveyMonkey	+	-	-	+	-
SurveyJS	+	-	+	+	+
Typeform	+	-	-	+	+

### 4.3.3 Core

В рамках самой библиотеки должны быть реализованы слои моделей и репозитория с сервисами.

Также, необходимо добавить в Core слой контроллеров, чтобы упростить интеграцию в сторонние проекты.

Таким образом, на уровне Core должны быть реализованы:

- модели и их хранение в базе данных;
- crud для моделей;
- операции с опросами;
- создание опроса;
- модификация опроса;
- прохождение опроса;
- получение результатов опроса;
- выгрузка результатов опроса;
- наборы интерфейсов для последующей интеграции;
- контроллеры.

### 4.3.4 Веб-интерфейс

Является лишь примером интегрирующего сервиса. Его первичная задача — продемонстрировать возможности интеграции и показать, как эту интеграцию провести.

Серверная часть должна представлять следующую функциональность:

- реализация User/Group, работа с пользователями;
- авторизация пользователей;
- реализовать весь функционал, который предлагает Core.

Клиентская часть должна представлять следующую функциональность:

- простейший менеджмент пользователей и назначение им прав;
- просмотр существующих опросов и их результатов;
- создание новых опросов;
- редактирование опросов;
- прохождение опросов;
- кнопка для выгрузки данных опроса в .xlsx формате.

страница «Создание опроса»:

- кнопка настройки опроса: при нажатии открывает окошко с параметрами: время начало опроса, время конец опроса, параметры видимости;

- кнопка добавить вопрос: при нажатии раскрывается формочка с выбором типом вопроса, варианты ответов;

- кнопка создать опрос: при нажатии страница переходит на страницу с результатами.

Страница «Результаты опроса»:

- страница для просмотра результатов и статистики опроса: держит постоянное подключение с помощью websocket для постоянного обновления результатов;
- общее количество респондентов: сколько людей приняло участие в опросе;
- процент завершения: какой процент людей завершил опрос относительно тех, кто начал его проходить;
- гистограммы и столбчатые диаграммы для вопросов с множественным выбором
- круговые диаграммы для визуализации процентного распределения ответов на вопросы с несколькими вариантами;
- облачко тегов для самых часто упоминаемых слов или фраз в ответах на открытые вопросы.

Страница «Опрос»:

Страница для проведения опроса. Содержит вопросы их поля для ответов.

#### **4.4 Этапы разработки**

Этапы разработки подразделяются на два вида контрольных пунктов. Первый вид контрольных пунктов — встреча с заказчиком каждую неделю и демонстрация процесса или результата за прошедшую неделю. Второй вид контрольных пунктов — контрольные точки с демонстрацией проделанной работы всей команды за длинный промежуток времени.

Первый тип контрольных точек очень частый и больше ориентирован для слежения процесса и анализа текущего направления разработки. Данный тип контрольных точек не нуждается в тестировании, так как часто имеет промежуточные результаты. Второй тип содержит три контрольных пункта с

длительными интервалами. Данный тип больше нуждается в тестировании, так как демонстрирует конкретные результаты команды.

#### **4.4.1 Первый контрольный пункт**

Текущий контрольный пункт является больше аналитическим и составляющим цели и задачи и проекта, и команды, а также общий концепт приложения с дизайном. К данному контрольному пункту команда провела аналитику, анализ конкурентов, техническое задание с usecases. А также была разработана уже большая часть дизайна будущего web-приложения. С результатами первого контрольного пункта можно ознакомиться в приложение А.

#### **4.4.2 Второй контрольный пункт**

Данный контрольный пункт уже содержит конкретные технические результаты разработки и дизайна. Дизайн был полностью готов и соответствовал прописанным usecases в техническом задании. Как и core библиотека, так и web-приложение имели функционал с авторизацией пользователей, созданием опросов и настройками опросов. Однако на данном этапе web-приложение еще не взаимодействовала с сервером и работала локально. При тестировании были выявлены небольшие недочеты с валидацией вводных данных при авторизации и регистрации. После обнаружения всех недочетов, они были устранены. С результатами второго контрольного пункта можно ознакомиться в приложение В.

#### **4.4.3 Третий контрольный пункт**

Третий контрольный пункт является заключительным и демонстрирует итоговый результат за весь процесс разработки. К текущему контрольному

пункту был доделан недостающий функционал у библиотеки: прохождение опроса и результаты. Также был представлен swagger, в последствии чего web-приложение имело возможность коммуницировать с сервером. Также были сделаны: страница прохождения опроса, результат опроса, просмотр пройденных опросов. При тестировании были обнаружены небольшие недочеты в интерфейсе, но они никак не мешали для пользователя получить главный и важный функционал сервиса. С результатами третьего контрольного пункта можно ознакомиться в приложение С.



## **ЗАКЛЮЧЕНИЕ**

Итоговый продукт полностью покрывает главные требования заказчика и пользователя, соответствует, как и основной заложенной идеи проекта, а именно: создание интегрируемой библиотеки для опросов вместе с демонстрирующим web-приложением функциональность библиотеки, так и визуальной составляющей, а именно: соответствие итога проекта с разработанным дизайном. Так как, web-приложение является демонстрацией главного функционала библиотеки и не имело четких требований от заказчика, функциональные и внешние требования были рассмотрены и разработаны самой командой.

При проведении тестирования были выявлены небольшое количество ошибок в UI составляющей и валидации вводных данных. Обнаруженные недочеты не мешают пользователю получить главный и необходимый функционал от сервиса.

Так как проект разрабатывается для заказчика, потребность в последующей разработке и поддержание присутствует. В первую очередь необходимо устранить все выявленные ошибки и недочеты, в последствие чего будет возможность добавления опциональных функций и доработок.

## ПРИЛОЖЕНИЕ А

### (обязательное)

#### Первый контрольный пункт

Рисунки 1, 2, 3 демонстрируют результаты работы к первому контрольному пункту.

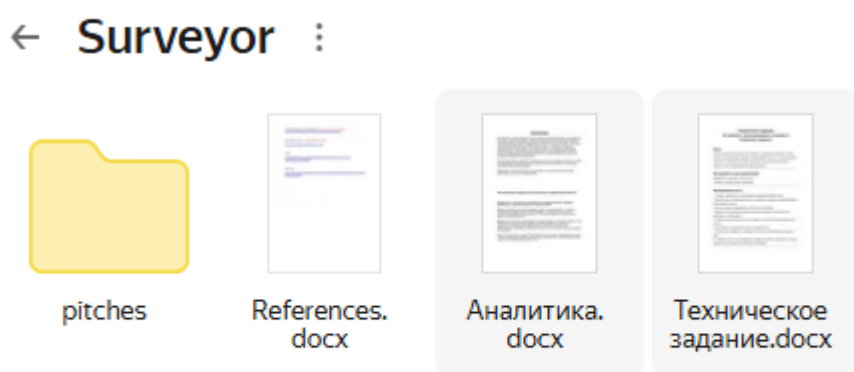


Рисунок 1 – Аналитика и техническое задание

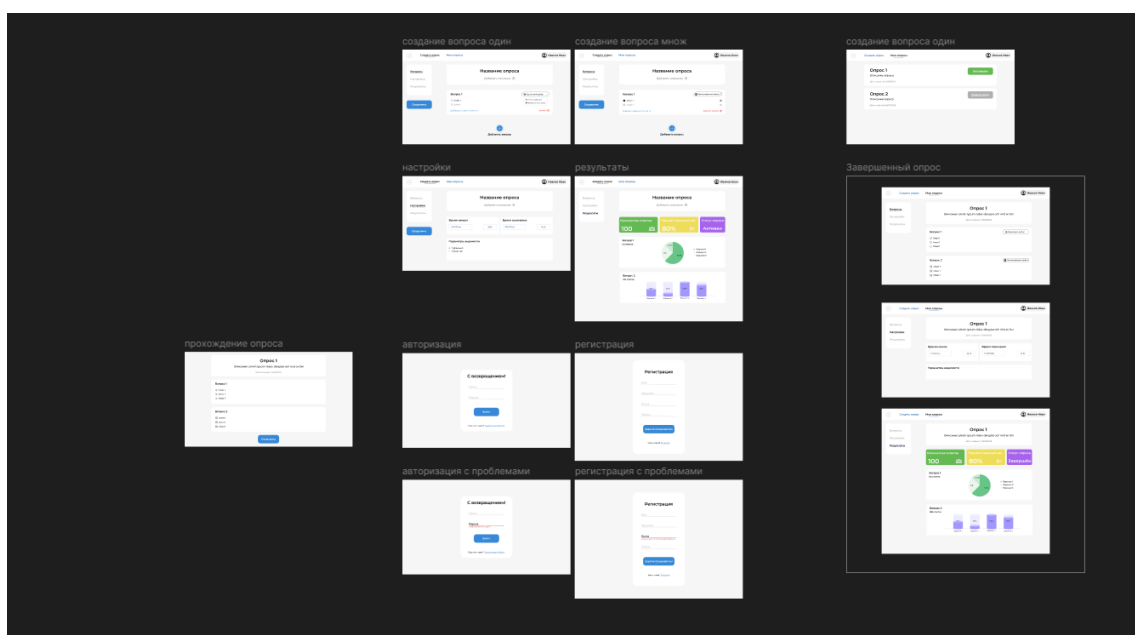


Рисунок 2 – Дизайн

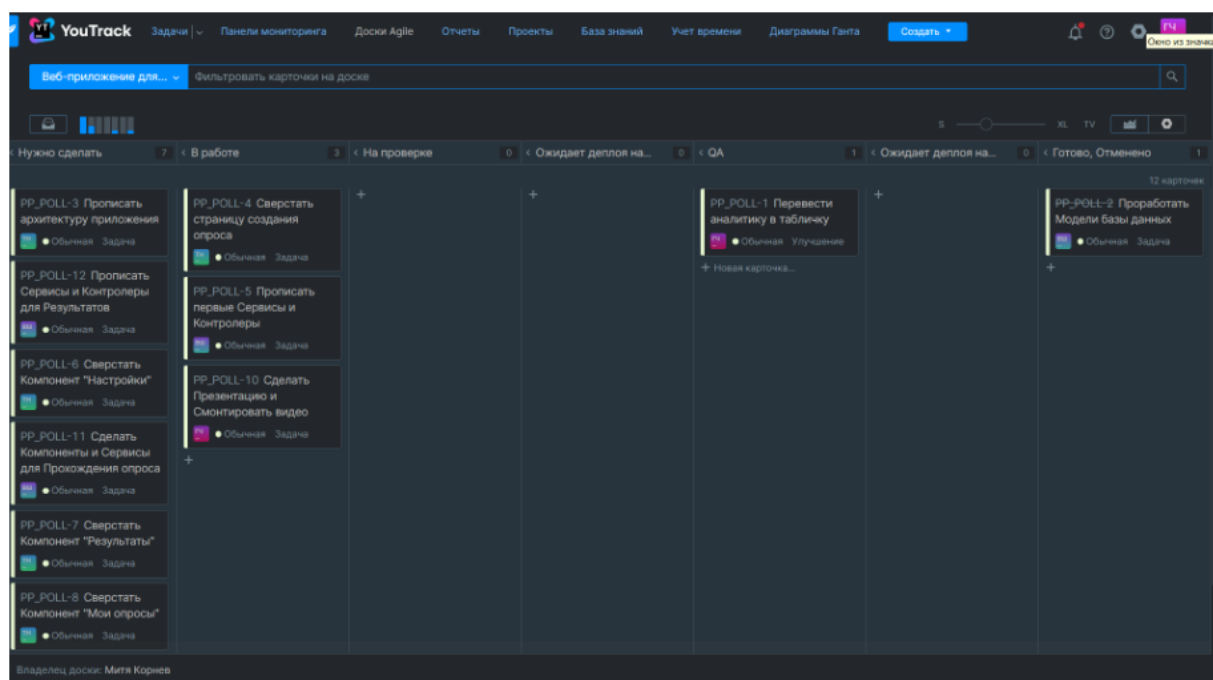


Рисунок 3 – YouTrack

## ПРИЛОЖЕНИЕ В

### (обязательное)

#### Второй контрольный пункт

Рисунки 4, 5, 6 демонстрируют результаты работы ко второму контрольному пункту.

Создать опрос   Мои опросы   Иванов Иван

Вопросы  
**Настройки**  
Результаты

Сохранить

**Название опроса**  
Добавить описание ⊕

**Время начала**  
mm/dd/yyyy   00:00

**Время окончания**  
mm/dd/yyyy   00:00

**Параметры видимости**

Рисунок 4 – Настройки опроса

Создать опрос   Мои опросы   Иванов Иван

Вопросы  
Настройки  
Результаты

Сохранить

Название опроса  
Добавить описание

Вопрос 1   Многократный выбор

Ответ 1  
Ответ 1

Добавить вариант ответа   Удалить вопрос

Добавить вопрос

Рисунок 5 – Создание опроса

## Регистрация

Имя

Фамилия

Почта

Пароль

Зарегистрироваться

Уже с нами? [Войдите!](#)

Рисунок 6 – Регистрация пользователя

## ПРИЛОЖЕНИЕ С

### (обязательное)

#### Третий контрольный пункт

Рисунки 7, 8, 9 демонстрируют результаты работы к третьему контрольному пункту.

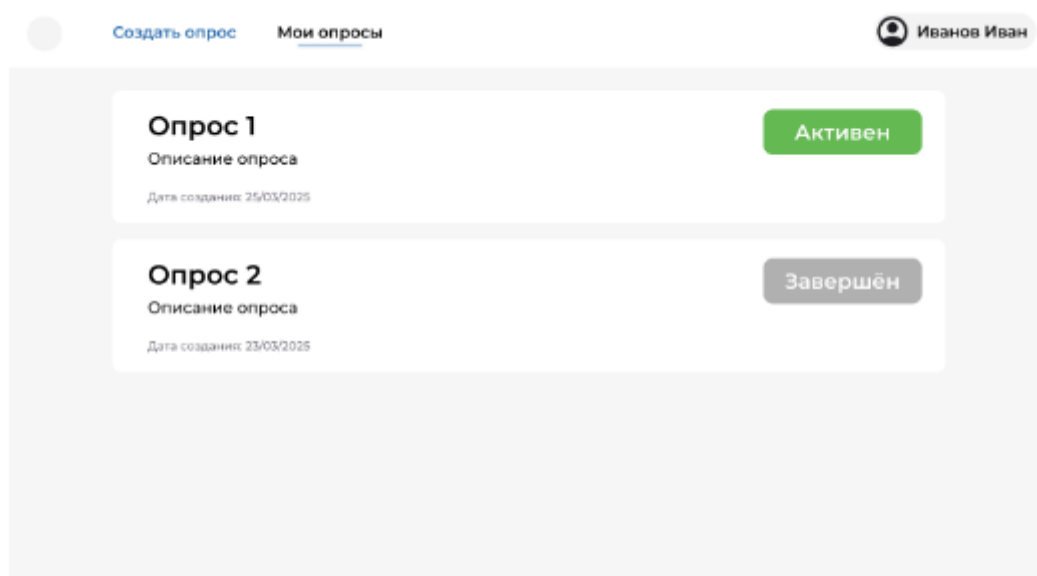


Рисунок 7 – Просмотр опросов

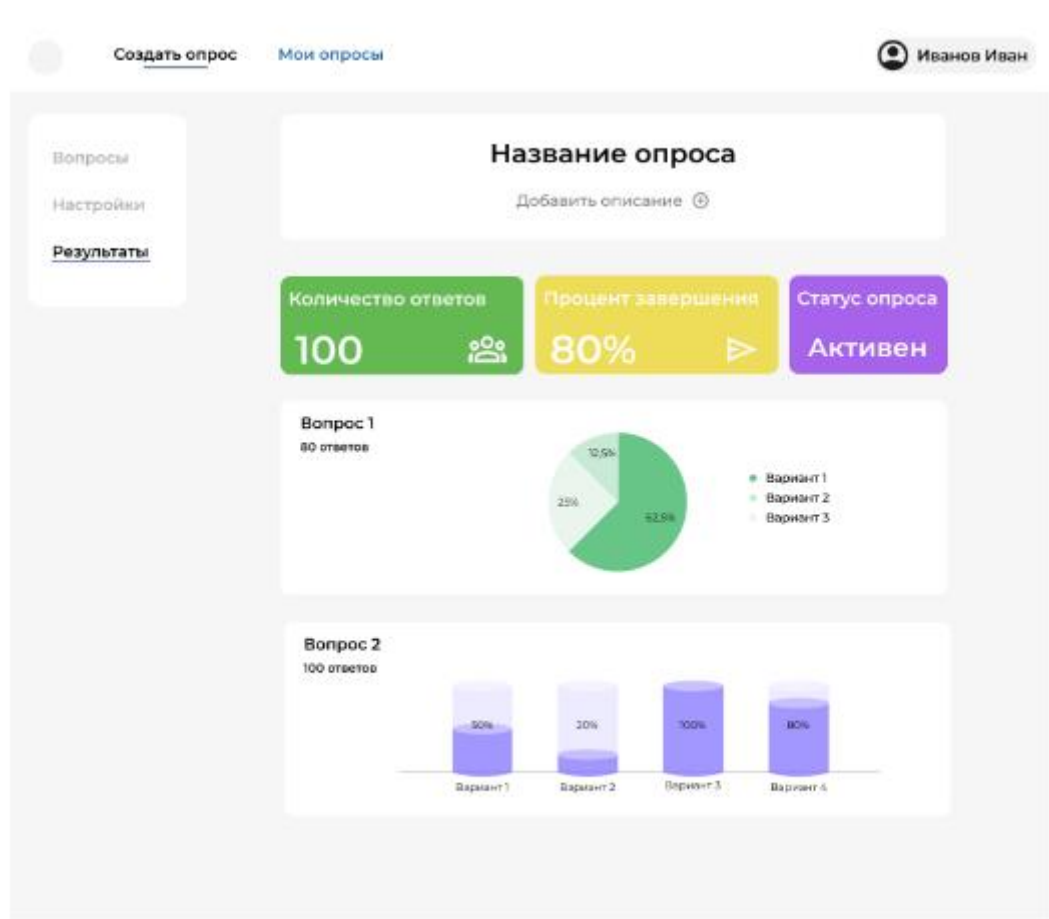


Рисунок 8 – Результаты опросов

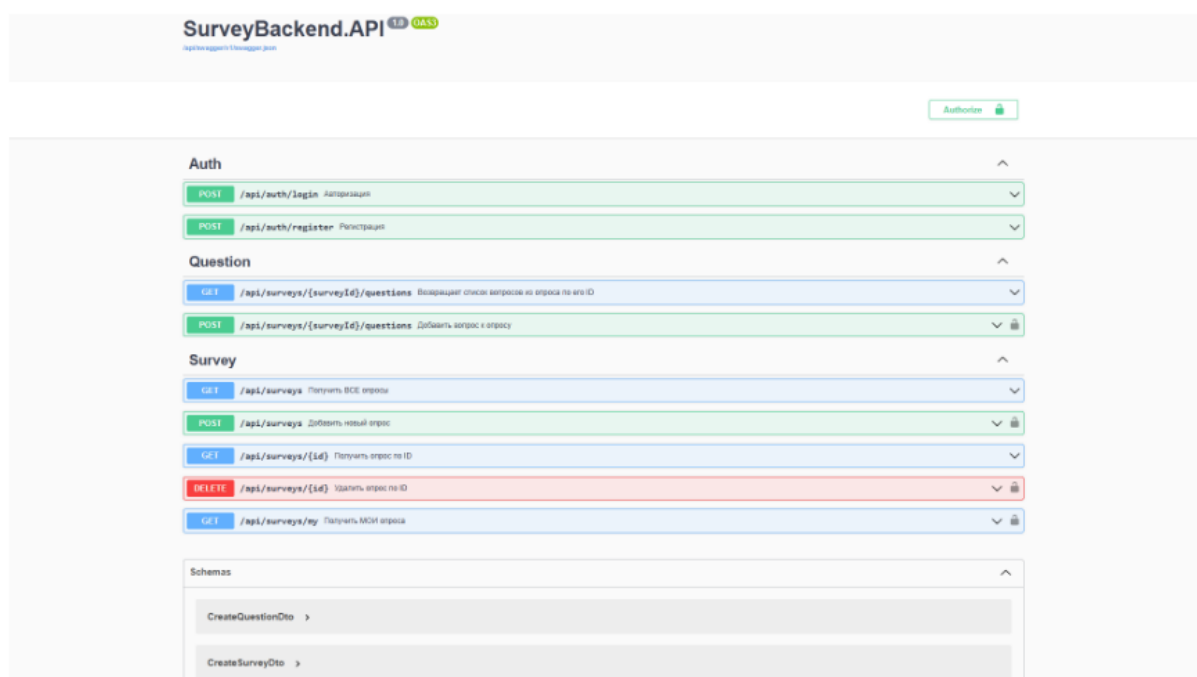


Рисунок 9 – Swagger