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

«Разработка веб-сервиса для оценки ПВК стажеров»

по дисциплине «Проектный практикум»

|  |  |
| --- | --- |
| Заказчик: Смирнов Д.С. |  |
| Куратор: Смирнов Д.С. |  |
| Студенты команды: Pixel Pioneers  Мельников М.Е. |  |
| Килязова Ю.С. |  |
| Гавриляк М.В. |  |
| Полякова Ю.А. |  |

Екатеринбург, 2025

СОДЕРЖАНИЕ

ВВЕДЕНИЕ 3

1 Основная часть 4

1.1 Разбор целевой аудитории 4

1.2 Обзор аналогов 4

1.3 Методология разработки и планирование деятельности 5

1.4 Архитектура программного продукта 6

1.4.1 Backend 6

1.4.2 Frontend 8

1.5 Аналитик – Килязова Юния Сергеевна 9

1.6 Дизайнер – Мельников Михаил Евгеньевич 14

1.7 Бэкенд разработчик – Гавриляк Михаил Витальевич 22

1.8 Фронтенд разработчик – Полякова Юлия Андреевна 24

ЗАКЛЮЧЕНИЕ 27

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 28

Приложение A (справочное) 30

ВВЕДЕНИЕ

Цель проекта: разработка веб-сервиса для взаимного оценивания студентов по методике «360» и другими видами оценивания с участием кураторов и экспертов.

Задачи:

1. Автоматизация процесса оценки профессионально-важных качеств,
2. Обеспечение прозрачности и удобства системы оценивания,
3. Интеграция с сервисом "Личный кабинет стажёра".

Актуальность проекта обусловлена недостатками существующего решения – веб-сервиса «Личный кабинет стажера», который использовался в течение 2022-2023 учебного года. Среди недостатков для кураторов — запутанное администрирование и отсутствие гибкости в добавлении этапов оценивания. Студенты же сталкиваются с отсутствием обратной связи и непонятной системой оценивания. Разработка нового сервиса позволит улучшить взаимодействие между кураторами и студентами, сделав процесс оценивания удобным и прозрачным, а также визуализировать результаты.

Область применения сервиса: образовательные учреждения, корпоративная среда, HR-среда и научные исследования.

Ожидаемыми результатами проекта является готовый веб-сервис с функционалом для кураторов, студентов и экспертов, а также документация по проекту.

1. Основная часть
   1. Разбор целевой аудитории

Целевая аудитория проекта: ЛРП(заказчик), кураторы, эксперты и студенты.

Ключевые проблемы, с которыми сталкивается заказчик, включают сложности с подбором стажёров и недостаток информации о soft skills. Ключевые проблемы студентов в свою очередь включают: отсутствие удобного сервиса, понятной системы оценивания и обратной связи. Проблемы кураторов и экспертов стоят где-то между заказчиком и студентами. Все эти проблемы препятствуют эффективному взаимодействию кураторов и студентов.

Для решения этих проблем предлагается веб-сервис, который обеспечит удобный и понятный процесс взаимного оценивания студентов друг другом, а также кураторами и приглашенными экспертами, с последующим отображением результатов.

* 1. Обзор аналогов

Сравнение аналогов приведено в таблице 1.

* + - 1. Сравнение аналогов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерий | **Edwica**[1] | **IT-Professions**[2] | **Potok**[3] | **Наш сервис** |
| **Оценка**  **hard skills** | Да | Да | Нет | Опционально |
| **Оценка**  **soft skills** | Да | Нет | Да | Да |
| **Метод оценки** | Автоматические тесты | Практические задания | Взаимное оценивание в команде | «360°» + экспертные оценки |
| **Дашборды и аналитика** | Да | Нет | Нет | Да |
| **Гибкость критериев** | Нет | Нет | Да | Да |
| **Обратная связь** | Нет | Нет | Базовая | Да |
| **Пример использования** | Подбор персонала | Оценка IT-скиллов | Внутренние рейтинги в стартапах | Проектные школы, отбор стажёров |

Проведённый анализ показал, что существующие решения охватывают лишь отдельные аспекты оценки профессиональных и личностных качеств стажёров, но не предлагают комплексного инструмента. Наш сервис закроет пробел в рынке, предложив универсальное решение для проектных школ с упором на объективность, удобство и аналитику.

* 1. Методология разработки и планирование деятельности

Разработка была поделена на три основных этапа: анализ и проектирование, разработка и тестирование. Был составлен бэклог задач, который был визуализирован с помощью диаграммы Ганта (Рисунок 1) [4]. Разработка велась по методологии Agile, еженедельно проводились встречи с куратором, на которых демонстрировалась выполненная работа за прошедшую неделю и обговаривались планы на следующую. На всех созвонах велся протокол [5].

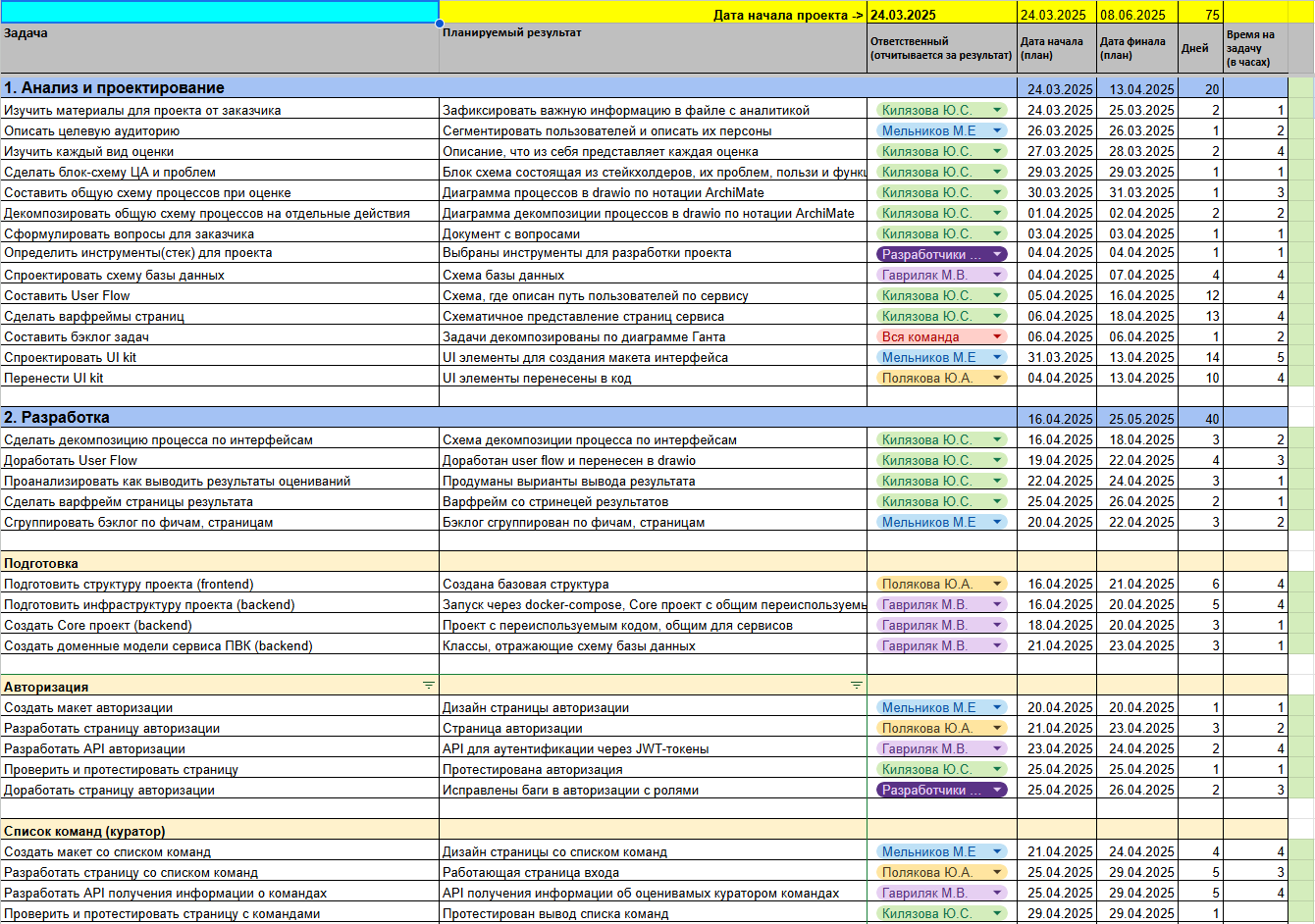


Диаграмма Ганта

В середине семестра был проведен созвон с заказчиком. Мы представили текущие наработки, рассказали о наших планах и видении проекта. Заказчик предоставил обратную связь, которая позволила скорректировать дальнейшую работу в соответствии с его ожиданиями.

Такая организация работы позволила нам двигаться поэтапно и эффективно справляться с задачами, обеспечивая качественное выполнение проекта на каждом этапе.

* 1. Архитектура программного продукта
     1. Backend

Разрабатываемый проект является частью более крупной экосистемы «Точка сбора», предназначенной для организации и управления стажировками и практиками. В рамках этой системы существует множество сервисов, включая IdentityService, сервис управления задачами, планировщик и CRM-системы. Наш проект представляет собой сервис оценки стажеров, который в перспективе должен быть интегрирован в общую экосистему.

В текущей реализации, учитывая ограничения доступа к существующим сервисам «Точки сбора», была разработана автономная версия системы оценивания. Архитектура проекта построена с учетом будущей интеграции с основной системой, но на данном этапе включает собственные упрощенные реализации необходимых компонентов.

Архитектурно проект разделен на несколько ключевых компонентов:

PIQ Service является основным сервисом системы, реализующим бизнес-логику оценивания. Он построен в соответствии с принципами Clean Architecture и DDD и включает следующие слои:

* API-слой для обработки HTTP-запросов посредством контроллеров,
* Application-слой, содержащий основную бизнес-логику,
* Infrastructure-слой для работы с внешними зависимостями,
* Models-слой с моделями сервиса.

Account Service в текущей реализации представляет собой упрощенную версию сервиса аутентификации. Он отвечает за генерацию JWT-токенов, в то время как PIQ Service осуществляет их валидацию на основе общих секретов. Такое решение является временным и предполагает замену на полноценную интеграцию с IdentityService основной системы в будущем.

Важной особенностью текущей реализации является то, что информация о мероприятиях, проектах, командах и участниках добавляется через механизм начального заполнения базы данных (seeding). Это решение принято исходя из понимания, что в полной версии системы такие данные будут поступать из соответствующих сервисов «Точки сбора».

Для обеспечения развертывания используется контейнеризация на основе Docker. Также в рамках другого предмета реализована terraform-спецификация деплоя всей инфраструктуры проекта, используя облачные сервисы Yandex Cloud.

Выбор микросервисной архитектуры и Clean Architecture обусловлен следующими факторами:

* необходимость будущей интеграции с существующей микросервисной системой,
* обеспечение независимости компонентов и их тестируемости,
* возможность независимого масштабирования различных частей системы,
* четкое разделение ответственности между слоями приложения.

Текущая архитектура, несмотря на некоторые временные решения, обеспечивает:

* демонстрацию основного функционала оценивания стажеров,
* возможность дальнейшей интеграции с экосистемой «Точки сбора»,
* масштабируемость и поддерживаемость кода,
* простоту развертывания через контейнеризацию.

Основной фокус в разработке был сделан на реализации бизнес-логики оценивания, так как это являлось приоритетной задачей для заказчика в этом семестре. При этом архитектура спроектирована таким образом, чтобы в будущем можно было легко заменить временные решения на полноценную интеграцию с существующими сервисами «Точки сбора».

* + 1. Frontend

Разработанный программный продукт представляет собой клиентскую часть веб-приложения, реализованную с использованием современного фреймворка Angular. Архитектура приложения построена на принципах компонентного подхода и разделения ответственности, что обеспечивает высокую поддерживаемость и масштабируемость системы.

Приложение организовано в соответствии с принципами модульной архитектуры Angular и включает следующие ключевые компоненты:

* корневой модуль приложения (app.component.ts) - точка входа в приложение, обеспечивающая инициализацию и конфигурацию основных сервисов,
* маршрутизация (app.routes.ts) - система навигации, реализующая паттерн Single Page Application,
* конфигурация приложения (app.config.ts) - централизованное управление настройками и зависимостями,
* компоненты пользовательского интерфейса (директория children/) - переиспользуемые UI-элементы,
* сервисы данных (директория data/) - слой взаимодействия с бэкенд-сервисами,
* перехватчики (директория interceptors/) - обработка HTTP-запросов и ответов,
* валидаторы (директория validators/) - бизнес-логика валидации данных.

Архитектура построена на следующих ключевых технологиях:

* Angular - основной фреймворк для разработки клиентской части,
* TypeScript - язык программирования, обеспечивающий строгую типизацию,
* SCSS - препроцессор CSS для стилизации компонентов,
* Docker - контейнеризация приложения для обеспечения изолированного выполнения.

Архитектурные решения:

1. Компонентная архитектура

Использование компонентной архитектуры обусловлено следующим:

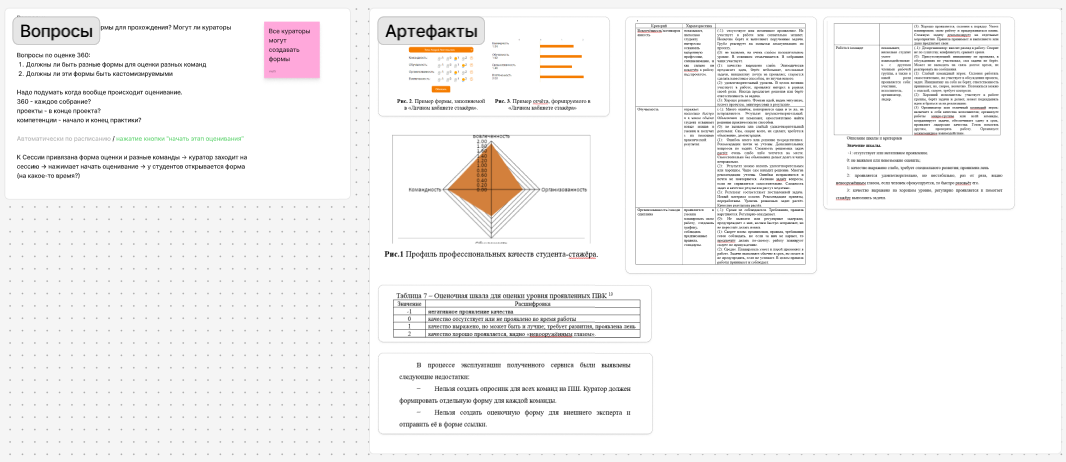
* каждый компонент является независимым с собственной логикой и представлением,
* переиспользование компонентов в различным частях приложения,
* четкое разделение ответственности между компонентами,
* упрощение тестирования за счет изоляции.

1. Контейнеризация

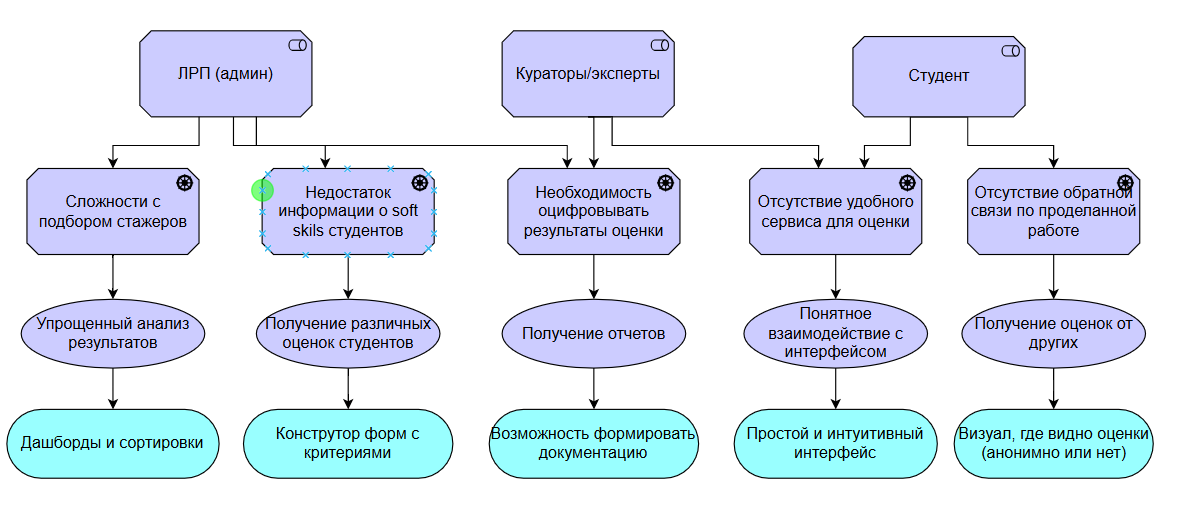
Использование Docker и Docker Compose обеспечивает:

* изоляцию окружения,
* упрощение развертывания,
* стандартизацию среды выполнения.
  1. Аналитик – Килязова Юния Сергеевна

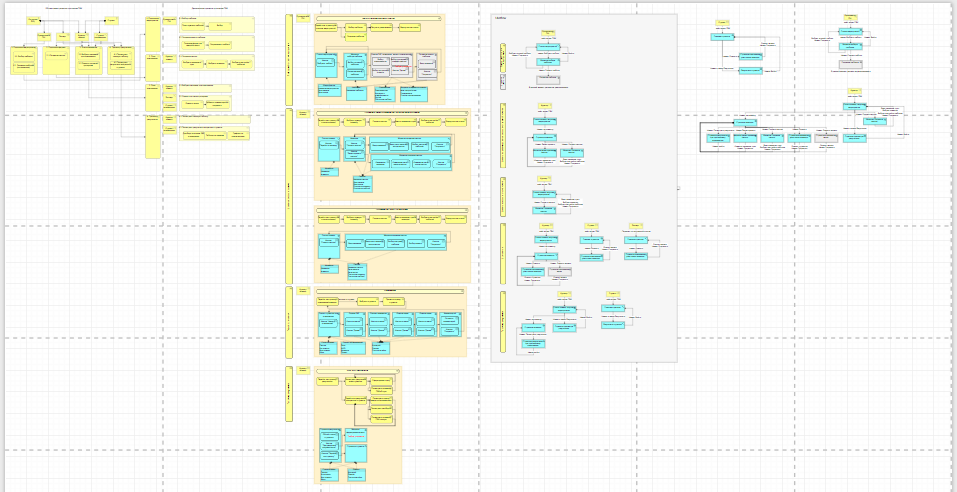
Работа над проектом началась с изучения предоставленных заказчиком материалов. Был проведён анализ артефактов и существующего решения: веб-сервиса «Личный кабинет стажёра» (Рисунок 2) [6]. Также были проанализированы проблемы целевой аудитории (Рисунок 3). На основе этого анализа были сформированы вопросы заказчику, а затем проведена работа по декомпозиции бизнес-процессов, описанию пользовательских интерфейсов и UserFlow на языке моделирования Архимейт, а затем переработаны из-за изменений в шаблоне от заказчика. (Рисунок 4) [7]. Также для удобства UserFlow был перенесен и в Figma (Рисунок 5).



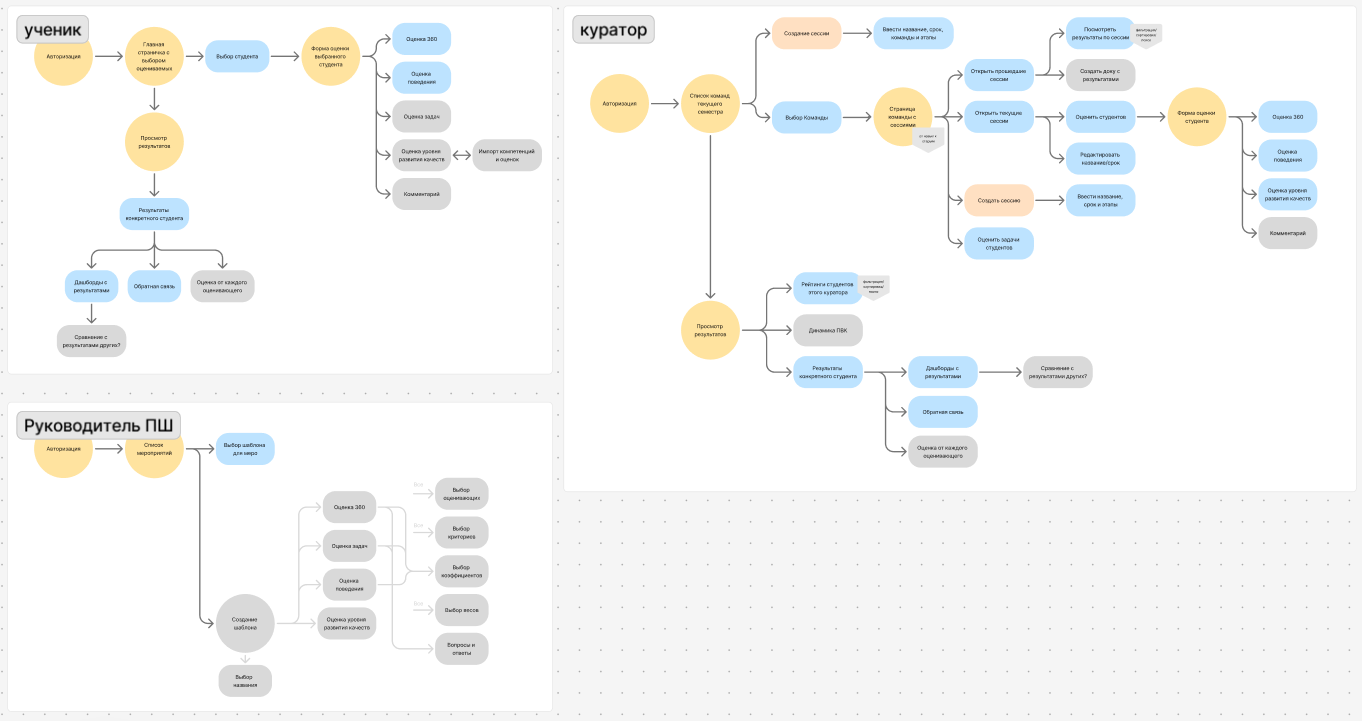
Артефакты



ЦА

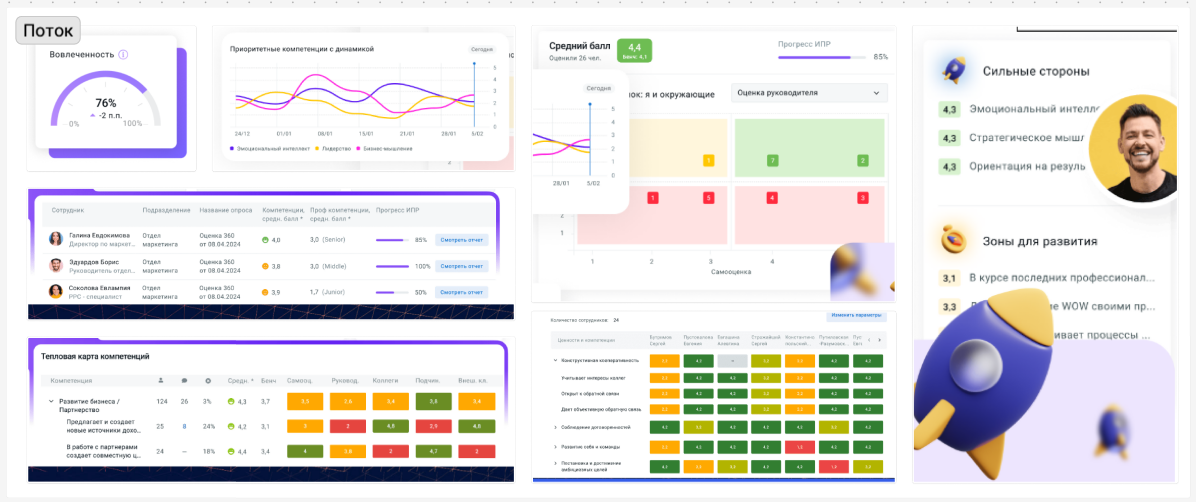


Бизнес-процессы в Архимейт

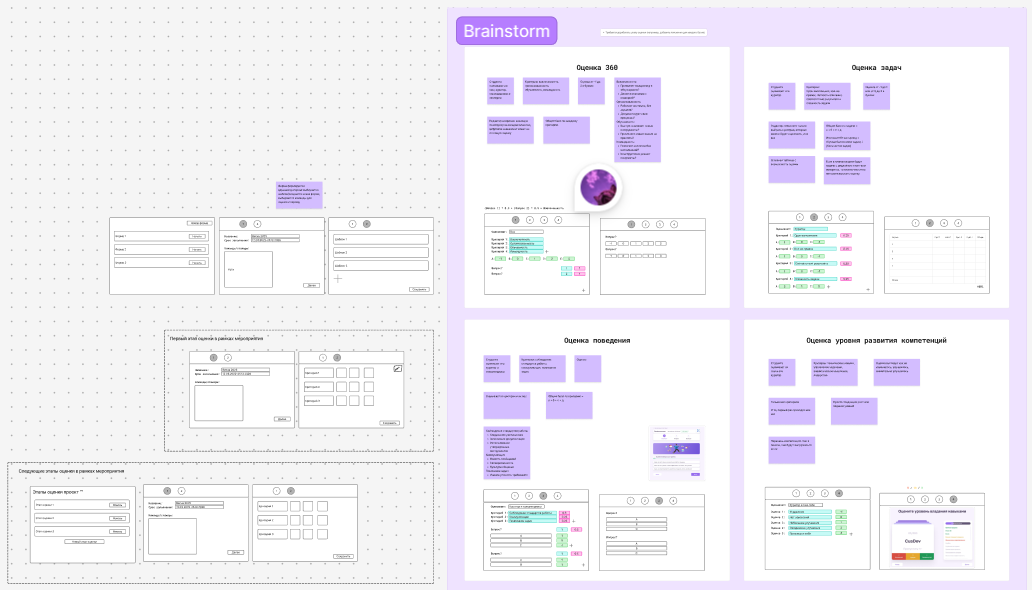


UserFlow в Figma

Были рассмотрены аналоги сервиса, основным вдохновителем для визу-ала стал сервис «Поток» (Рисунок 6). Была проведена аналитика по оцениванию ПВК: виды оценивания, критерии и шкала оценивания, вайрфреймы страниц (Рисунок 7) [8]. Затем был проведен обмен вопросами с заказчиком, который даже включал видео и файл [9] [10]. Все вопросы объединены в приложении А.

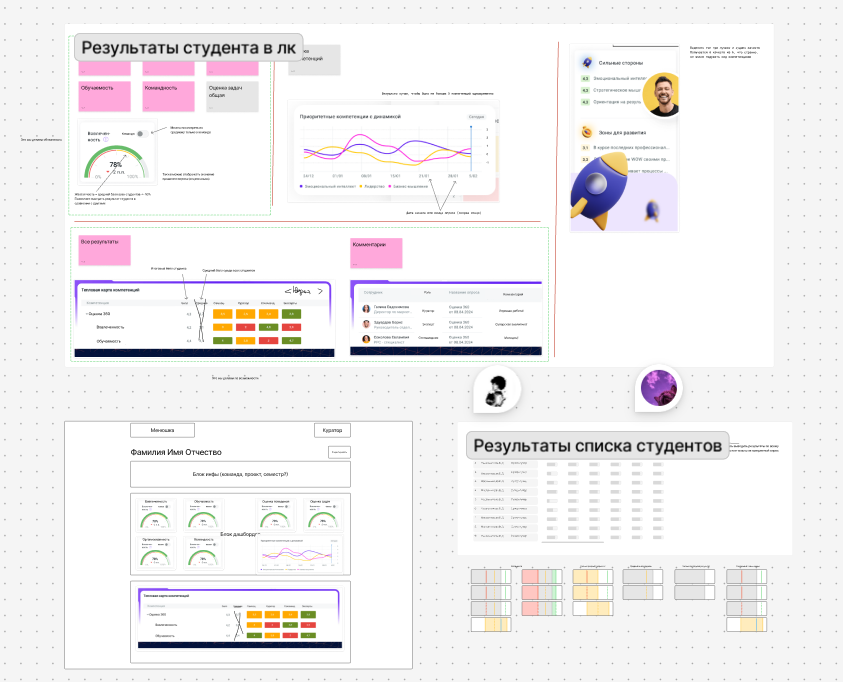


Разбор аналога «Поток»



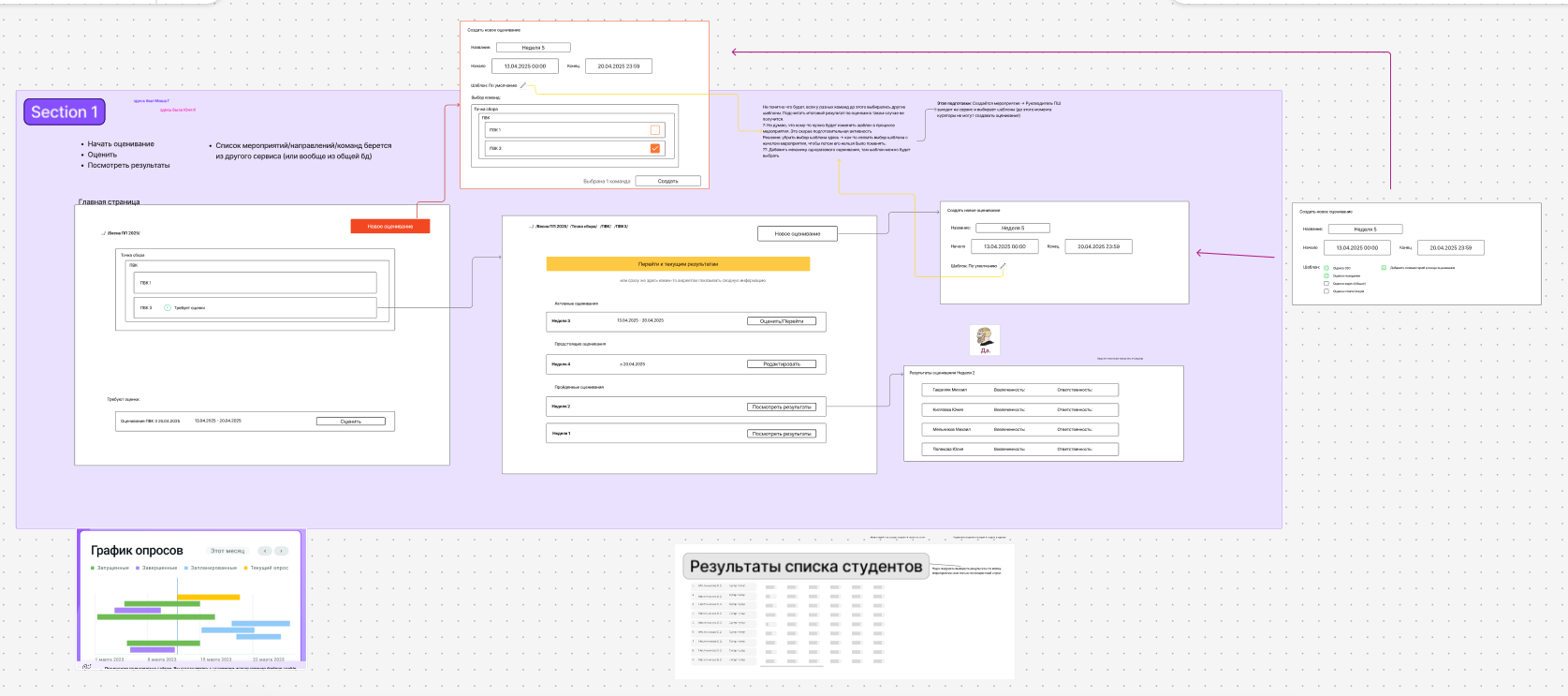
Аналитика оценивания

Также была проведена аналитика по визуальному отображению результатов. Были выделены основные элементы и сформированы вайрфреймы страницы результатов (Рисунок 8).



Аналитика результатов

Совместно с бэкендером были созданы вайрфреймы главной страницы, которые стали основой для будущего макета (Рисунок 9). Также постоянно велось обсуждение по поводу макета с командой.

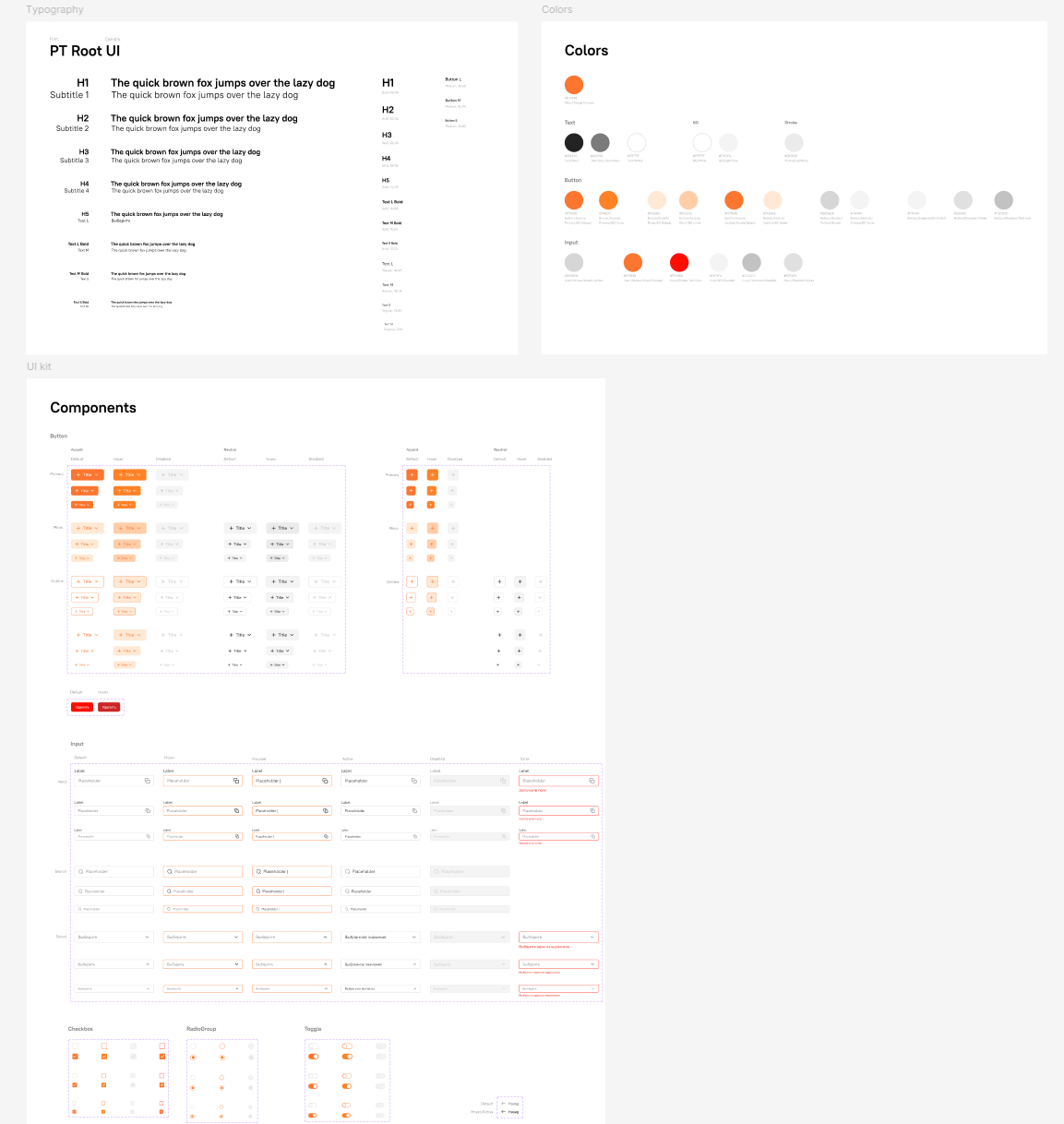


Вайрфреймы главной страницы

В завершение были расписаны конкретные вопросы для шаблона оценивания оценки 360 и поведения, включая описание и шкалу оценивания для каждого критерия [11] [12].

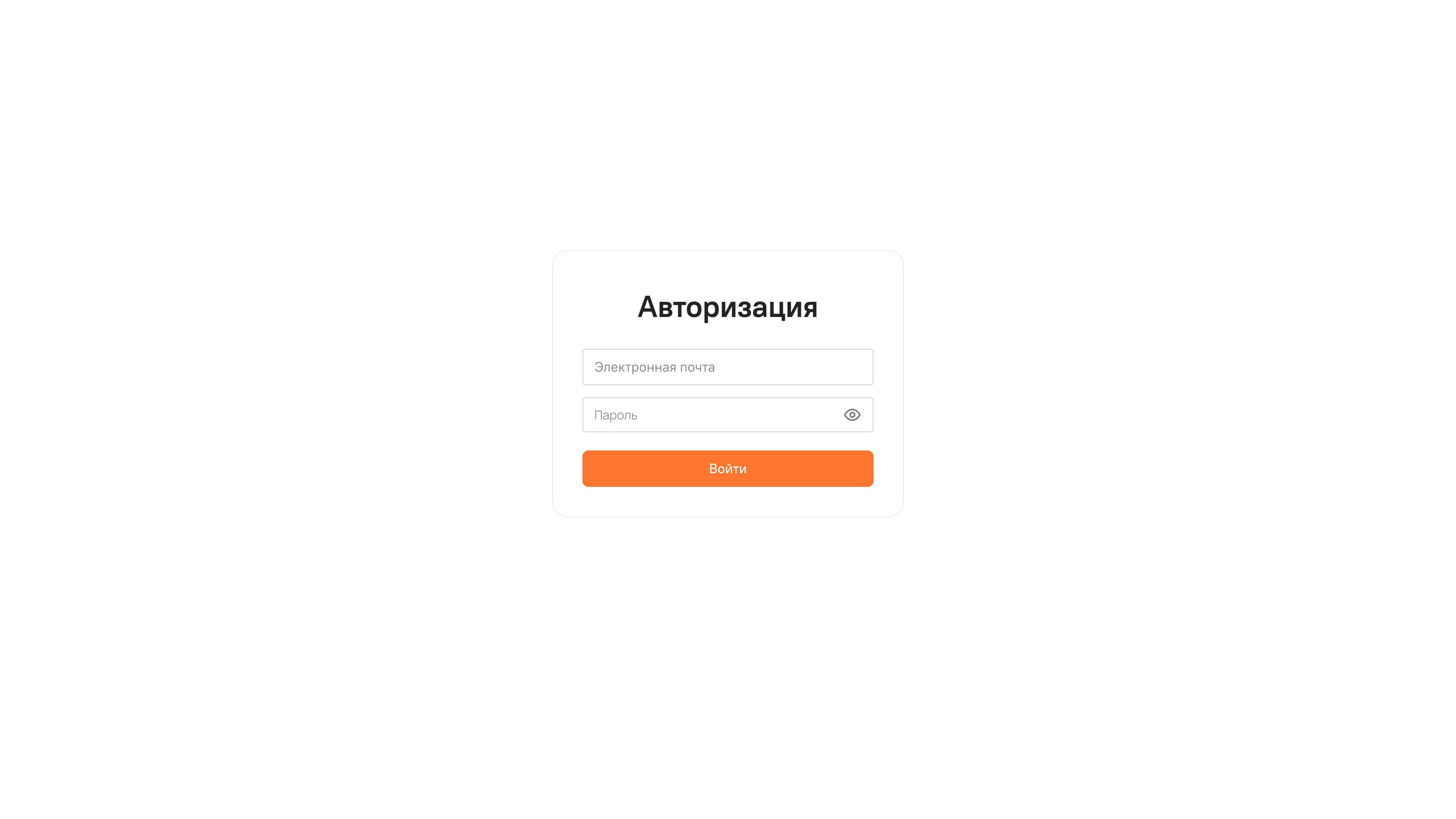
* 1. Дизайнер – Мельников Михаил Евгеньевич

Работа над дизайном интерфейса сервиса [13] началась с создания визуальной базы проекта. Был собран UI kit с базовыми компонентами, построена типографика и определена цветовая палитра интерфейса (Рисунок 10). Все это заложило основу для будущего дизайна.



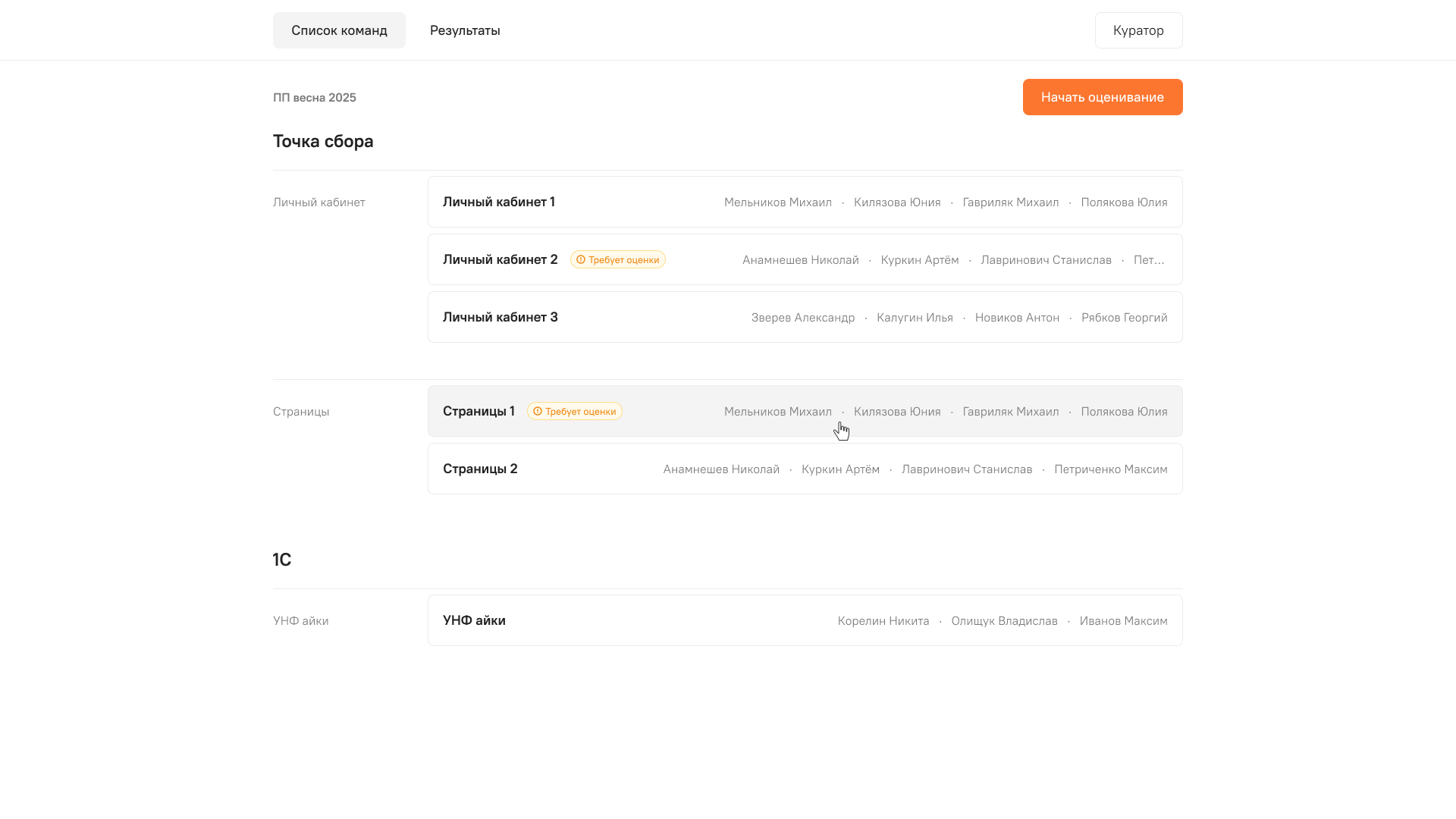
UI kit

Работа с отрисовкой интерфейса началась с макета авторизации (Рисунок 11).



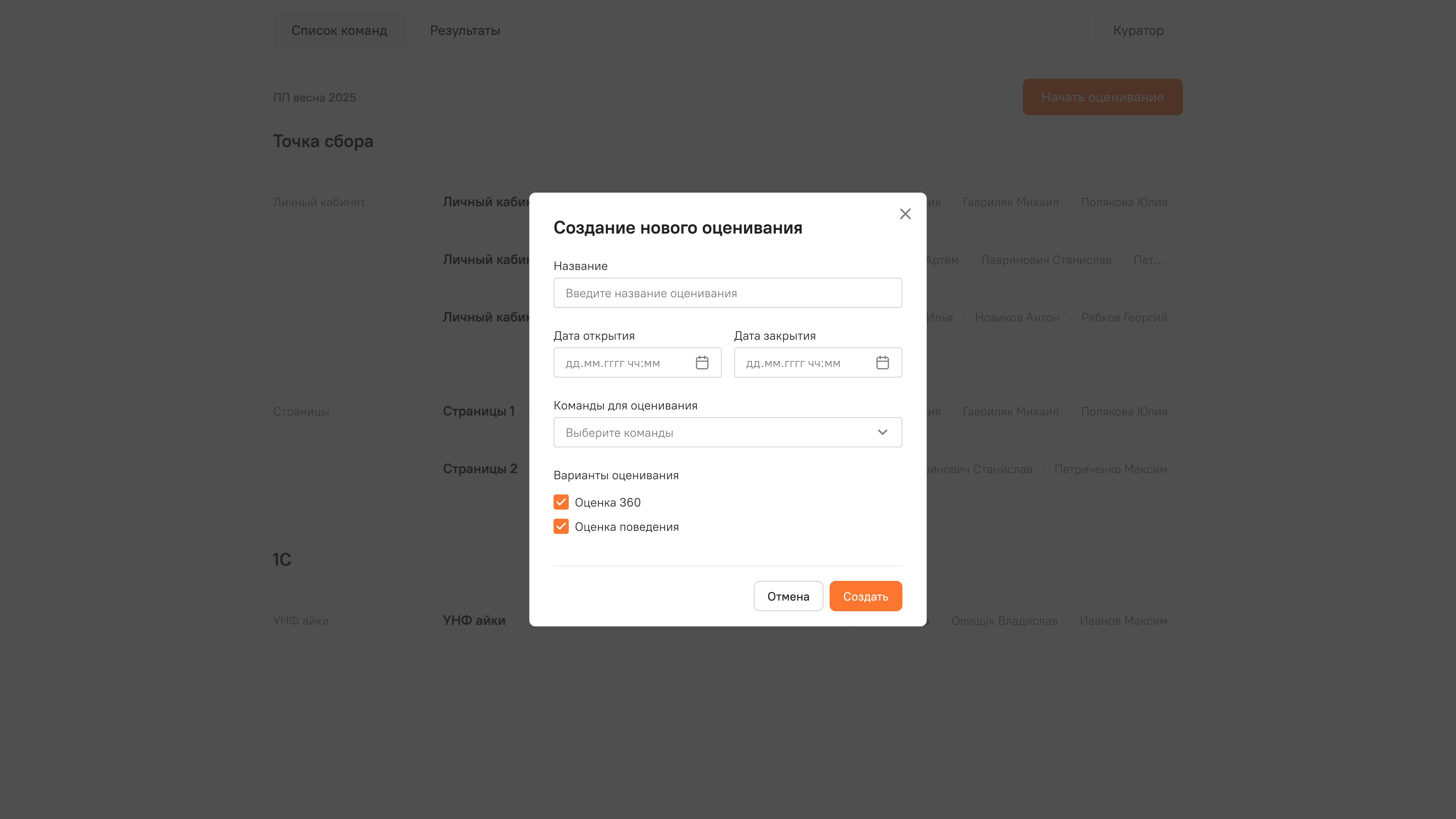
Авторизация

Далее началась работа над аккаунтом куратора. Была отрисована главная страница со списком команд (Рисунок 12), которые ведет куратор. Также была проработана «шапка» сервиса, где находится основная навигация.



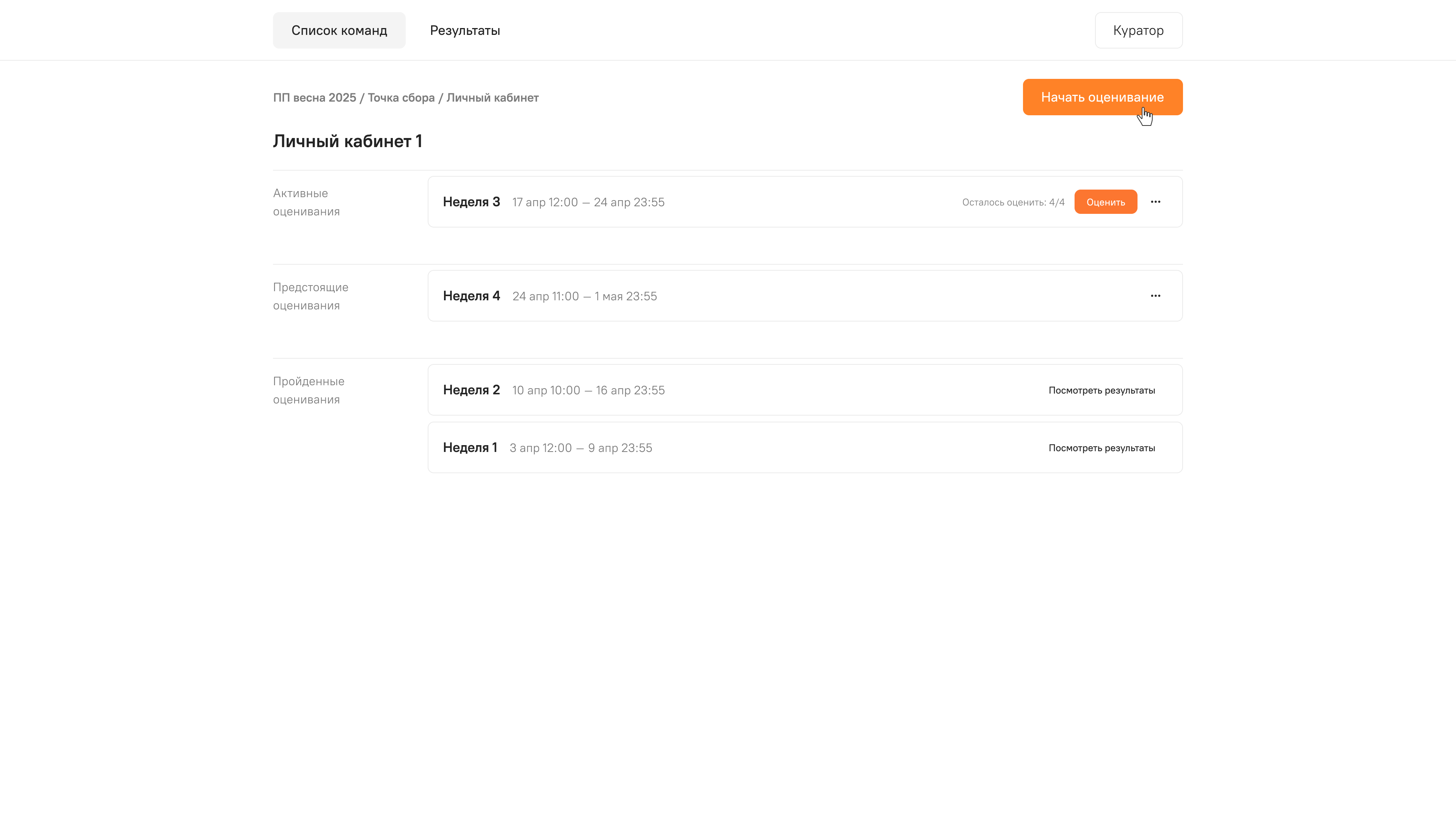
Список команд

Также был разработан функционал по созданию оценивания сразу нескольким командам. При нажатии на кнопку «Начать оценивание» на главной, открывалось модальное окно (Рисунок 13) с настройкой создания оценивания.



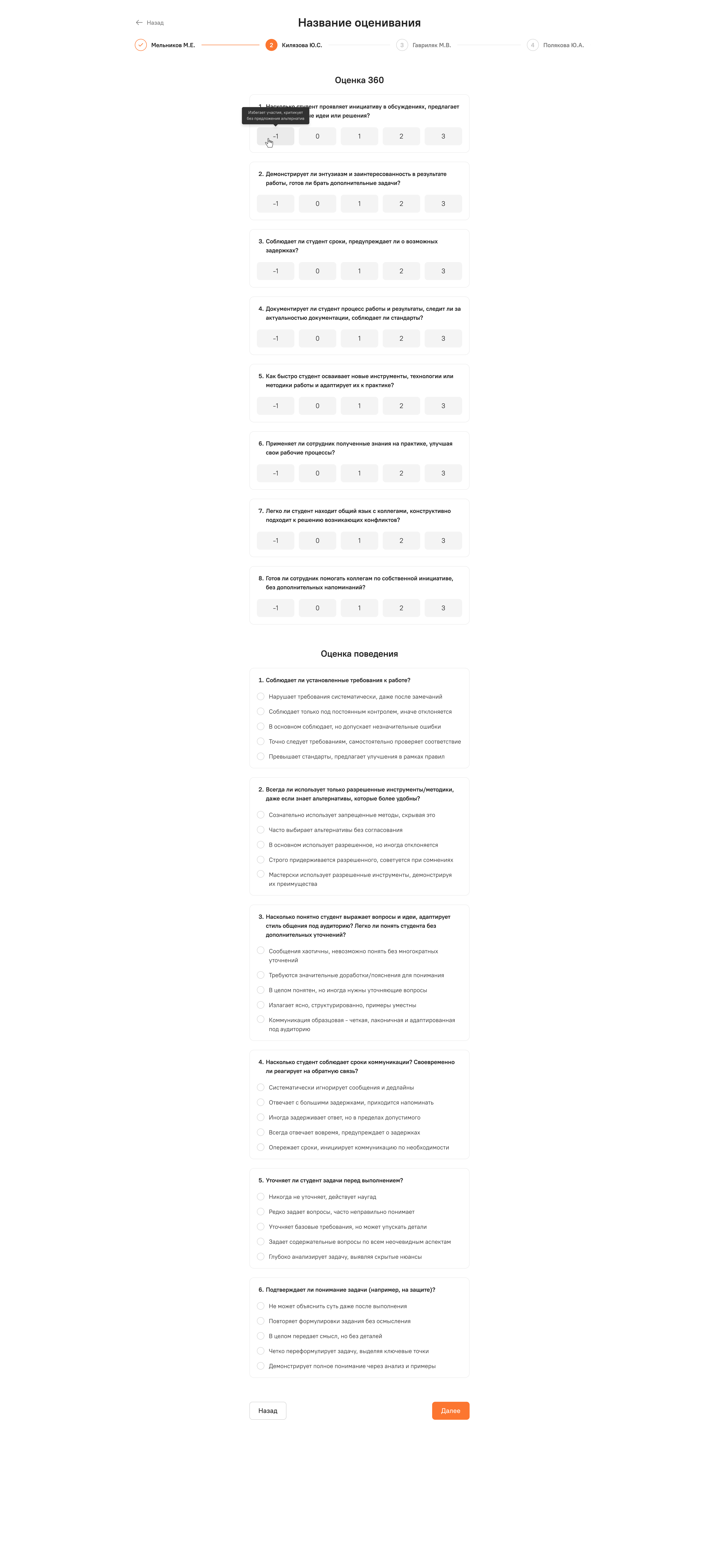
Модальное окно создания оценивания

Далее была отрисована страница конкретной команды и ее список оцениваний (Рисунок 14). Здесь отображается активное оценивание, которое доступно прямо сейчас и в него можно перейти. Запланированное оценивание, если куратор заранее создал, но дата открытия оценивания еще не наступила. И пройденные оценивания, которые уже прошли. Также можно редактировать оценивания: изменять названия, сроки или вовсе удалить.



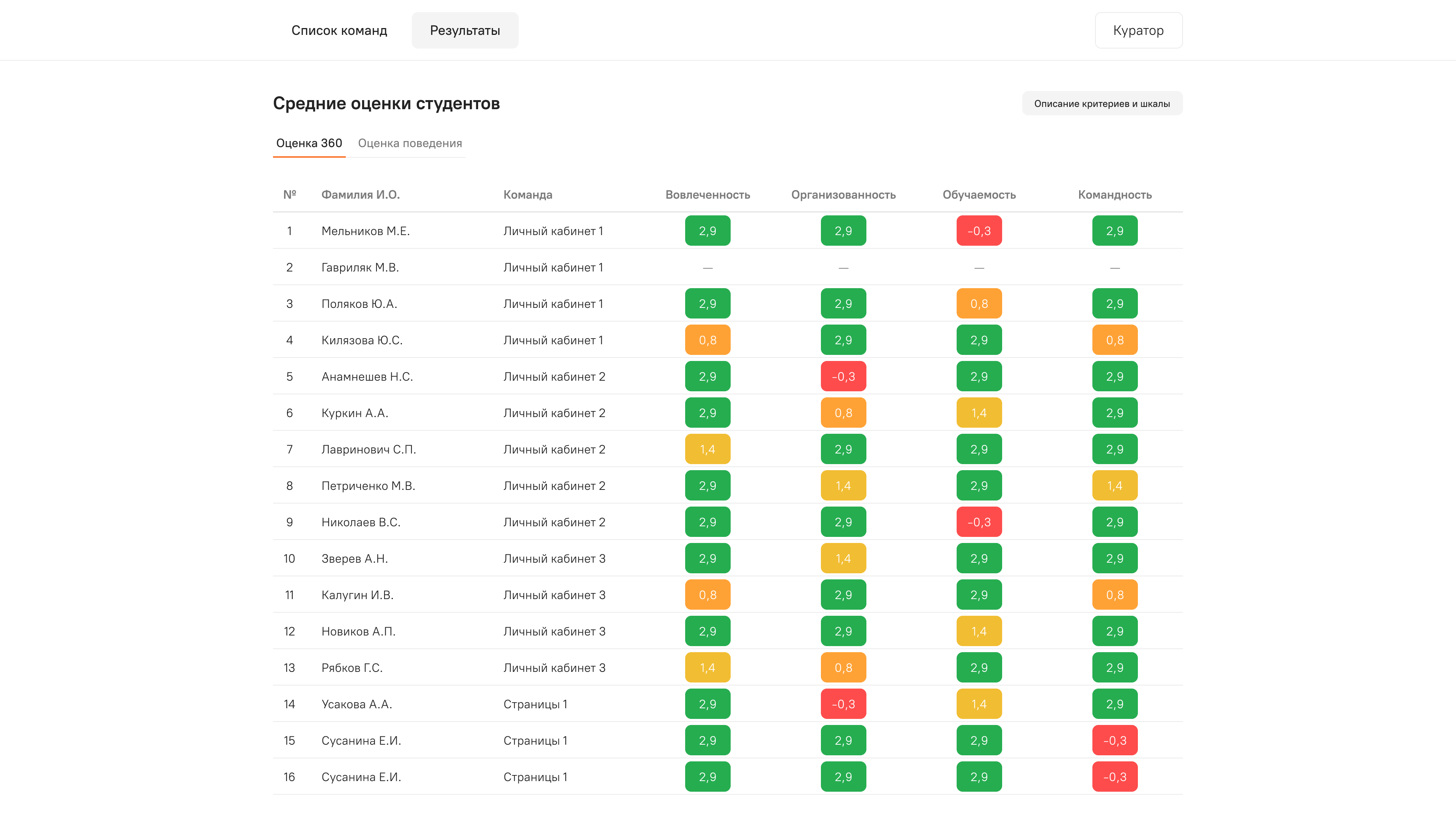
Оценивания конкретной команды

Если нажать на кнопку «Оценить» в активном оценивании, то мы попадаем на форму оценивания (Рисунок 15), которую проходят все студенты, которые участвую в данном оценивании и их куратор. В форме в зависимости от настроек при создании, может быть только один или сразу оба вида оценивания при прохождении. Оценка «360» представляет из себя 8 вопрос, по 2 вопроса на каждый критерий, который мы оцениваем. Варианты ответа в данном оценивании представляют из себя цифры, при наведении на которые появляется подсказка, что значит эта цифра. Оценка «Поведения» это 6 вопрос, но также по 2 вопроса на каждый критерий. В ней ответы представлены в текстовом варианте.

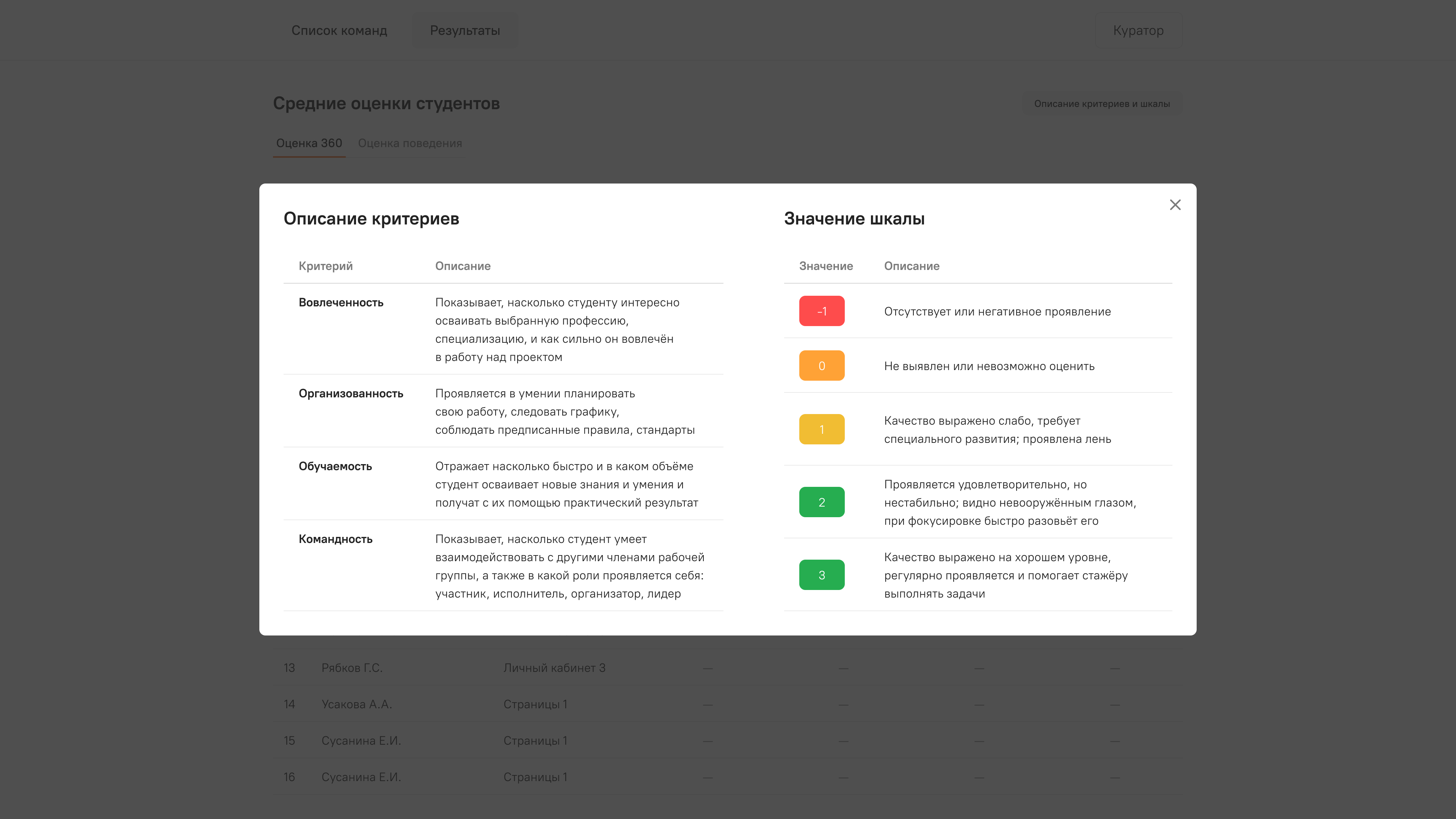


Форма оценивания

Следующим была отрисована страница результатов средней оценки по всем студентам (Рисунок 16). Здесь куратор может просматривать средние оценки студентов по каждому критерию. Также у куратора имеется подсказка в виде модального окна (Рисунок 17), в котором написано описание каждого критерия и описание, что значат баллы.

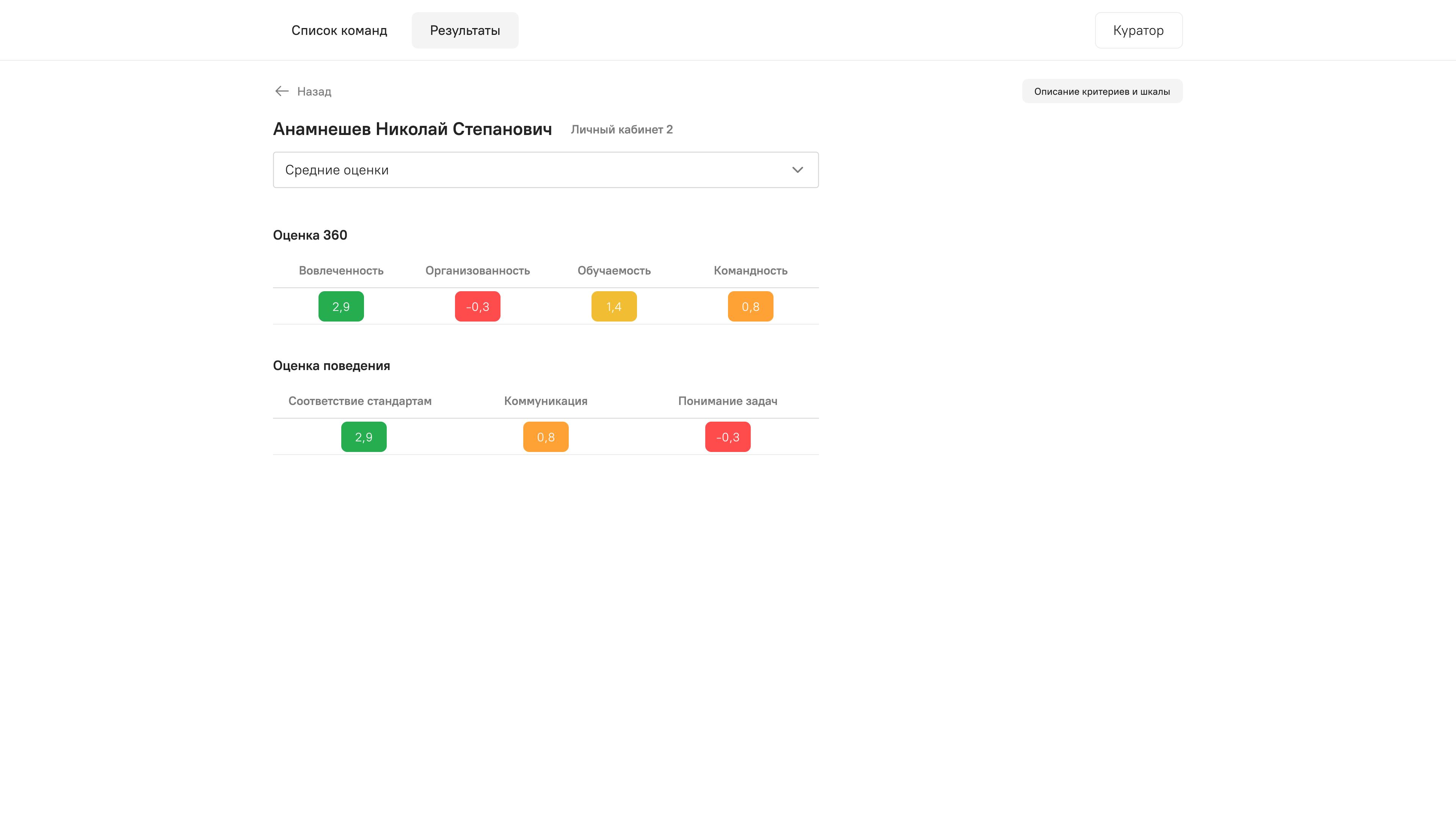


Результаты студентов



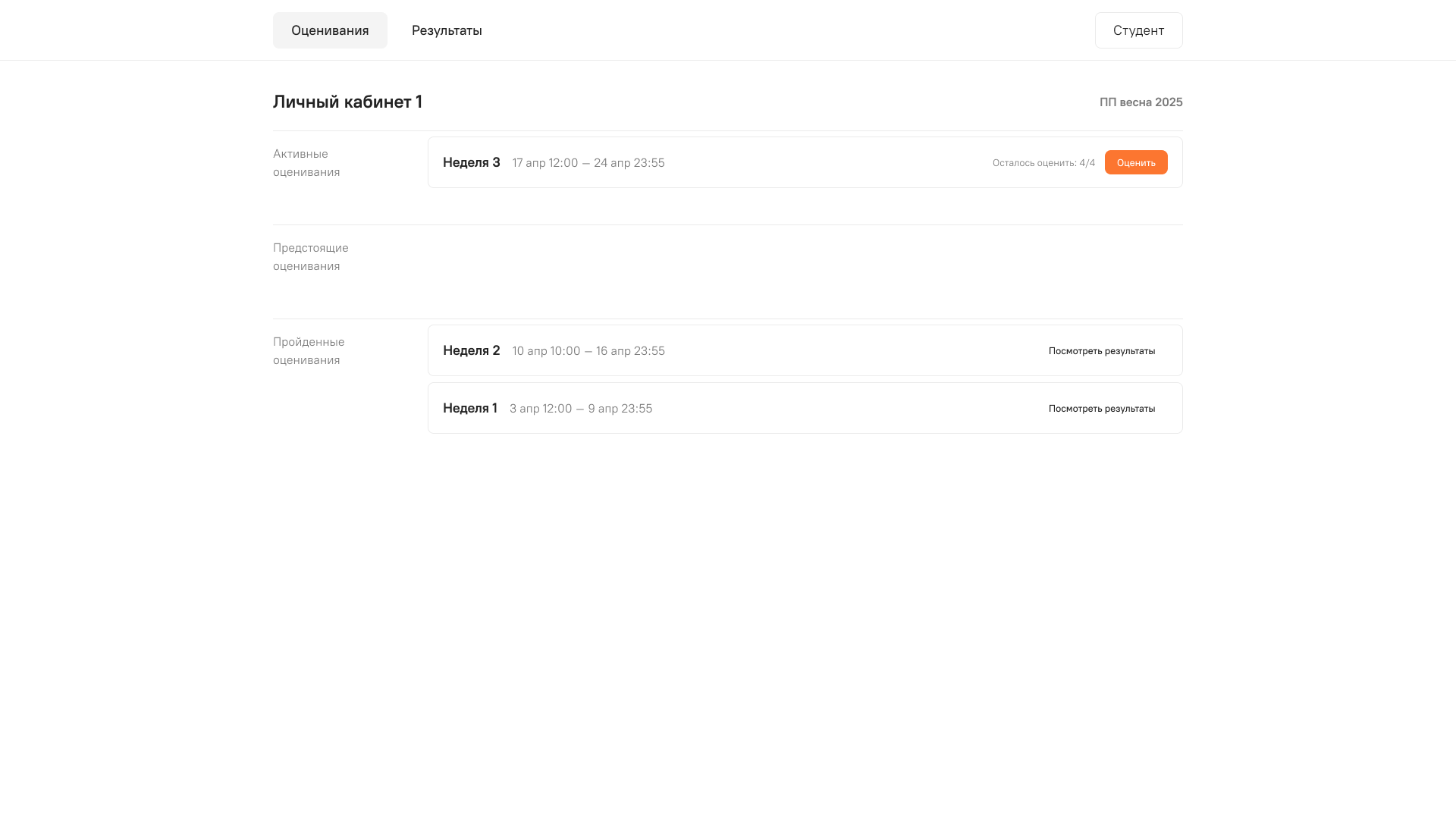
Подсказка с описанием критериев и шкалы

Помимо этого, была отрисована страница результатов по конкретному студенту (Рисунок 18). Куратор может перейти на конкретного студента и увидеть его средние оценки по каждому критерию или же посмотреть оценку за конкретное оценивание, которое когда-либо проходило.



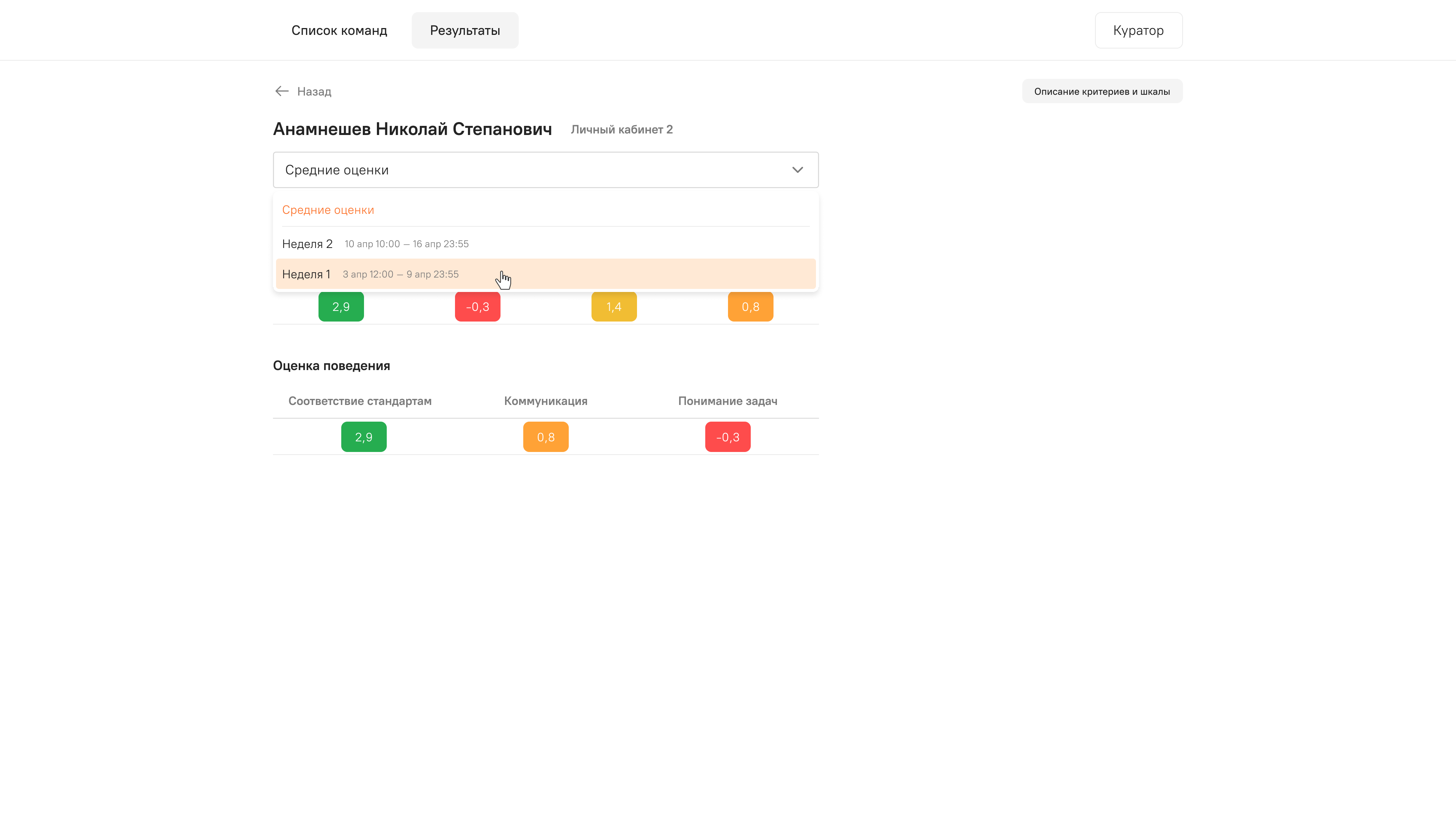
Результаты по конкретному студенту

В аккаунте ученика была отрисована главная страница, на которой отображается список оцениваний его команды (Рисунок 19).



Список оцениваний

По нажатию «оценить» студент попадает на форму с оцениванием, которую уже ранее рассмотрели у куратора. Также у студента есть возможность просматривать свои средние баллы за оценивания или баллы за конкретное оценивание (Рисунок 20).



Результаты студента

* 1. Бэкенд разработчик – Гавриляк Михаил Витальевич

В рамках разработки backend-части проекта [14] была реализована микросервисная архитектура на платформе .NET, обеспечивающая решение для проведения комплексной оценки участников команд.

Ключевым компонентом системы является PIQ Service, реализующий основную бизнес-логику оценивания. Сервис поддерживает два основных типа оценивания: оценку 360 градусов и поведенческую оценку. Оценка «360» позволяет участникам команды оценивать друг друга по различным профессионально-важным качествам, в то время как поведенческая оценка фокусируется на конкретных поведенческих индикаторах, определенных для каждой роли в команде.

Система ролей реализована на уровне API с использованием JWT-токенов и включает три основные роли:

* администратор: полный доступ ко всем функциям системы, включая управление оцениваниями и просмотр результатов всех участников,
* куратор: возможность создавать оценивания для подопечных команд, просмотр результатов своих команд,
* участник команды: доступ к прохождению назначенных оцениваний и просмотру собственных результатов.

Бизнес-логика системы включает алгоритмы обработки результатов оценивания:

* расчет средних показателей по каждому критерию оценки,
* агрегация результатов для формирования общего профиля компетенций,
* валидация корректности введенных данных,
* формирование итоговых отчетов по результатам оценивания.

API сервиса построено с использованием REST-архитектуры и включает следующие основные эндпоинты:

* управление оцениваниями (создание, получение списка доступных оцениваний),
* работа с формами оценивания (получение формы, сохранение результатов),
* получение результатов (индивидуальных и командных),
* получение информации о командах и участниках.

Система хранения данных включает следующие основные сущности:

* информация о командах и их участниках,
* критерии оценивания для различных типов оценок,
* результаты проведенных оцениваний,
* связи между оцениваемыми и оценивающими.

Для обеспечения безопасности реализованы:

* аутентификация на основе JWT-токенов,
* базовое разграничение доступа на уровне ролей,
* логирование основных действий пользователей,
* валидация входных данных на уровне API.

Особенности реализации бизнес-логики:

* поддержка различных типов вопросов в формах оценивания (числовые оценки, текстовые комментарии),
* гибкая система критериев оценивания, позволяющая настраивать различные наборы компетенций,
* возможность проведения как индивидуальных, так и командных оцениваний,
* формирование сводных отчетов по результатам оценивания.

В текущей реализации система интегрируется с упрощенным сервисом аутентификации (Account Service), который отвечает за генерацию JWT-токенов. PIQ Service осуществляет валидацию этих токенов на основе общих секретов. Такое решение является временным и предполагает замену на полноценную интеграцию с IdentityService основной системы «Точка сбора» в будущем.

Данные о мероприятиях, проектах, командах и участниках в текущей версии добавляются через механизм начального заполнения базы данных, что также является временным решением до интеграции с соответствующими сервисами основной системы.

В целом, backend-часть системы представляет собой функциональное решение, обеспечивающее основные возможности для проведения оценки участников команд. Архитектура системы спроектирована с учетом будущей интеграции в экосистему «Точка сбора» и позволяет расширять функциональность по мере необходимости.

* 1. Фронтенд разработчик – Полякова Юлия Андреевна

В рамках разработки фронтенд-части проекта [15] было реализовано веб-приложение на базе Angular, обеспечивающее комплексное решение для проведения и анализа оценок командной работы. Основные функциональные возможности включают:

Система авторизации и управления доступом реализована с использованием JWT-токенов, обеспечивая безопасный вход пользователей и хранение их сессий. Реализована система ролей с тремя уровнями доступа: администратор, куратор и участник команды. Для каждой роли созданы соответствующие guard-компоненты, контролирующие доступ к различным разделам приложения.

Маршрутизация приложения построена с учетом ролевой модели, где для каждой роли доступны соответствующие разделы. Для администраторов и кураторов реализован доступ к управлению оцениваниями, прохождению оцениваний и просмотру результатов, в то время как участники команд имеют доступ только к своей команде, прохождению оцениваний и собственным результатам.

В разделе команд реализован функционал просмотра списка команд с информацией об их составе и текущих оцениваниях. Для администраторов и кураторов доступно создание новых оцениваний как для отдельных команд, так и для нескольких команд одновременно. При создании оценивания можно выбрать типы оценок: оценка «360» или оценка «поведения».

Система оценивания включает два основных типа форм: оценка «360» и оценка «поведения». Каждый тип оценки имеет свой набор критериев и шкалу оценивания. Реализован пошаговый процесс заполнения форм с возможностью сохранения промежуточных результатов и возврата к ним позже. Для каждого участника команды создается отдельная форма оценки, что обеспечивает независимость оценок.

В разделе результатов реализована система анализа и визуализации полученных оценок. Для администраторов и кураторов доступен общий список результатов всех студентов с возможностью фильтрации по типу оценки. Реализована детальная страница результатов конкретного студента, где можно просмотреть его оценки по всем критериям как в общем виде, так и по отдельным оцениваниям.

Интерфейс приложения разработан с учетом современных требований к пользовательскому опыту, включая интуитивно понятную навигацию и информативные визуальные элементы. Реализована система индикации статуса оценок с использованием цветовой кодировки для быстрого восприятия результатов.

Техническая реализация включает использование современных практик разработки на Angular, включая реактивное программирование с использованием RxJS, компонентный подход, сервисную архитектуру и систему интерцепторов для обработки HTTP-запросов. Реализована система обработки ошибок и состояний загрузки для обеспечения стабильной работы приложения.

В целом, фронтенд-часть системы представляет собой полноценное решение для проведения и анализа командных оценок, обеспечивающее удобный интерфейс для всех участников процесса и надежную техническую реализацию.

ЗАКЛЮЧЕНИЕ

Разработанный веб-сервис для взаимного оценивания ПВК студентов соответствует базовым требованиям заказчика. Сервис предоставляет удобный интерфейс для кураторов, студентов и экспертов, а также визуализирует результаты, что решает ключевые проблемы целевой аудитории: отсутствие обратной связи, запутанность администрирования и непонятную систему оценивания.

Качество продукта оценивается на уровне демонстрационной версии т.к. интеграция с экосистемой «Точка сбора» не была реализована.

Предложения по улучшению:

* интеграция с «Точкой сбора»,
* реализация других видов оценивания и обратной связи,
* реализация личного кабинета стажёра,
* создание конструктора шаблонов оценивания.

Проект успешно реализован на уровне прототипа для данного на разработку времени. Текущая версия сервиса демонстрирует концепцию и будет удовлетворять базовые требования при использовании, но не является завершенным продуктом. Дальнейшая доработка позволит превратить его в надежный инструмент для оценки компетенций.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Аналог «Edwica»: сайт. – 2025. – *URL*: https://edwica.ru/ (дата обращения: 27.05.2025).
2. Аналог «Найти в IT»: сайт. – 2025. - *URL*: https://it-professions.ru/(дата обращения: 27.05.2025).
3. Аналог «Поток Оценка 360»: сайт. – 2025. - *URL*: https://potok.io/ (дата обращения: 27.05.2025).
4. Гугл-таблица с декомпозицией задач: сайт. – 2025. – *URL*: https://docs.google.com/spreadsheets/d/1Gg1POgi5mz2N7WwmA-PnsEizoEgkDwsC/edit?usp=sharing&ouid=118069554956047564724&rtpof=true&sd=true (дата обращения: 27.05.2025).
5. Протоколы собраний: сайт. – 2025. – *URL*: https://docs.google.com/document/d/1z8QvIEpPQSrEP7rD76JAcADf4efl6AFzJdK4QtWscfM/edit?usp=sharing(дата обращения: 27.05.2025).
6. Фигма с аналитикой: сайт. – 2025. – *URL*: https://www.figma.com/board/q9mdeCSxd9XpioJFmUMjwV/MeetPoint-PIQ-%D0%B0%D0%BD%D0%B0%D0%BB%D0%B8%D1%82%D0%B8%D0%BA%D0%B0?node-id=0-1&t=d0ypOOfEpwrjH403-1(дата обращения: 27.05.2025).
7. Draw.io с аналитикой: сайт. – 2025. – *URL*: https://drive.google.com/file/d/1mh5Ifhs2VmxIK93QETID\_Ve-sJ7h2Ggi/view?usp=sharing(дата обращения: 27.05.2025).
8. Фигма с аналитикой оценок: сайт. – 2025. – *URL*: https://www.figma.com/board/7I4S0E3Vq7S0hQsixi27NR/%D0%9E%D1%86%D0%B5%D0%BD%D0%BA%D0%B8?node-id=0-1&t=D6VHXicQRDDAhN7Q-1 (дата обращения: 27.05.2025).
9. Видео с вопросами заказчику: сайт. – 2025. – *URL*: https://drive.google.com/file/d/1Z2imrV8naorYdUjpDsMLq-6uCObMpWJh/view?usp=sharing(дата обращения: 27.05.2025).
10. Значение шкалы: сайт. – 2025. – *URL*: https://docs.google.com/document/d/12SHn3KFIKxbkdYgJlC5XFco88rojFjDzKpND266LE1w/edit?usp=sharing(дата обращения: 27.05.2025).
11. Шаблон оценка 360: сайт. – 2025. – *URL*: https://docs.google.com/document/d/1INaxwzH-AzQY1Gs\_tx3wjgylZ8KWDjRm/edit?usp=sharing&ouid=118069554956047564724&rtpof=true&sd=true(дата обращения: 27.05.2025).
12. Шаблон оценка поведения: сайт. – 2025. – *URL*: https://docs.google.com/document/d/1nATqs\_cjkuYmwYqfpokoJpBvJNyQnEMX/edit?usp=sharing&ouid=118069554956047564724&rtpof=true&sd=true(дата обращения: 27.05.2025).
13. Фигма с дизайном интерфейса: сайт. – 2025. – *URL*: https://www.figma.com/design/y40pwlAnDnVuhzcFBbw1CK/MeetPoint-PIQ-%D0%B4%D0%B8%D0%B7%D0%B0%D0%B9%D0%BD?node-id=0-1&t=maKII6i3YTol5HU5-1(дата обращения: 27.05.2025).
14. Гитхаб backend: сайт. – 2025. – *URL*: https://github.com/AnTaif/meetpoint-PIQ-backend(дата обращения: 27.05.2025).
15. Гитхаб frontend: сайт. – 2025. – *URL*: https://github.com/1zbbxzak1/meetpoint-PIQ-frontend(дата обращения: 27.05.2025).
16. (справочное)
17. Кого вы подразумеваете под экспертом?
18. В каком варианте оценивания он нужен? (Оценка 360, оценка по задачам, поведенческая оценка?)
19. Может ли куратор создавать опросы для команд, которые он не курирует?
20. Куратор сказала, что опрос должен быть каждую неделю. Является ли целесообразным повторять каждую неделю опрос, включающий оценку 360, поведения, задач и компетенций? Или нужно, чтобы еженедельный опрос включал только оценку 360 и поведения? Есть ли смысл оценивать компетенции каждую неделю? Есть предложение вынести оценку задач в отдельную вкладку, т.к. задачи оценивает только куратор, чтобы куратор мог оценить их в любое время вне опроса.
21. Куратор сказала, что хочет запускать опрос после собрания с командой. Тогда придется запускать опрос отдельно для каждой команды, потому что собрания у нас в разные дни. Не логичнее запускать не сразу после собрания, а в какой-то день сразу для некоторого количества команд?