**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ 4**](#_Toc179487093)

[**1 СИСТЕМНЫЙ АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 5**](#_Toc179487094)

[1.1 Анализ предметной области 5](#_Toc179487095)

[**2 ПРОЕКТИРОВАНИЕ 8**](#_Toc179487098)

[2.1 Общая концепция системы 8](#_Toc179487099)

[2.2 Выбор технологий и инструментов 9](#_Toc179487100)

[**3 РАЗРАБОТКА 11**](#_Toc179487103)

[3.1 Алгоритмы обработки данных: 11](#_Toc179487104)

[3.2 Реализация интерфейса и взаимодействия 14](#_Toc179487105)

[**4 ТЕСТИРОВАНИЕ 16**](#_Toc179487106)

[4.1 Тестирование 16](#_Toc179487107)

[**ЗАКЛЮЧЕНИЕ 22**](#_Toc179487108)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 23**](#_Toc179487109)

**ПРИЛОЖЕНИЕ А. ТЕКСТ ПРОГРАММЫ**

# ВВЕДЕНИЕ

Выбор подходящего учебного направления – один из ключевых этапов в жизни каждого студента, от которого зависит его будущая профессиональная реализация. В последние годы профориентационные системы, предлагающие студентам и школьникам помощь в определении своей профессиональной траектории, получили широкое распространение. Такие системы позволяют учащимся оценить свои профессиональные склонности и интересы, выявить сильные и слабые стороны, и, исходя из этих данных, выбрать наиболее подходящее направление обучения. Однако традиционные подходы к профориентации часто оказываются недостаточно точными, поскольку базируются на ограниченных и обобщенных методиках оценки, которые не учитывают индивидуальные особенности каждого студента.

С появлением и развитием технологий глубокого обучения и языковых моделей (LLM), возможности профориентационного тестирования значительно расширились. Использование LLM позволяет анализировать ответы пользователей на более глубоком уровне, выявляя тонкие аспекты их профессиональных предпочтений и личных качеств. Применение LLM делает возможным формирование точного профиля пользователя, сопоставимого с особенностями различных образовательных программ и факультетов, что позволяет повысить точность рекомендаций.

В данном проекте представлена разработка системы профориентационного тестирования с применением LLM, которая нацелена на помощь в выборе одного из пяти факультетов Брестского государственного технического университета. Система реализована в виде веб-приложения, где пользователи проходят тест из 25 вопросов. Ответы анализируются с помощью локальной LLM, что позволяет более точно определить подходящий факультет, а также предоставляет подробные объяснения результатов, позволяя пользователю понять, как его ответы связаны с полученной рекомендацией.

# 1 СИСТЕМНЫЙ АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

**1.1 Анализ предметной области**

Профориентационное тестирование направлено на помощь школьникам и студентам в выборе учебного направления и будущей карьеры. Основная цель такого тестирования – выявление предпочтений, способностей и личностных качеств человека, которые соответствуют определённым профессиям и специальностям. Правильно подобранное направление обучения позволяет раскрыть потенциал студента и увеличить его шансы на успешное трудоустройство в будущем.

Существуют различные методы профориентационного тестирования, включающие психологические опросники, анкеты на выявление интересов и тесты на оценку профессиональных склонностей. Эти подходы ориентированы на анализ таких качеств, как аналитическое и креативное мышление, коммуникативные навыки, склонность к работе в команде, уровень физической активности и другие важные характеристики.

Классические методы профориентационного тестирования включают в себя проверенные временем подходы и инструменты, разработанные для выявления интересов, способностей и склонностей человека к определённым профессиональным сферам. Вот основные из них:

Психологические опросники и анкеты на выявление интересов. Опросники, такие как методики Джона Холланда или опросники о профессиональных интересах, позволяют определить профессиональные предпочтения человека, то есть сферы, к которым он испытывает особый интерес. В основе таких опросников лежит предположение, что человеку легче развиваться и быть успешным в деятельности, которая его увлекает. Например, результаты тестов Холланда распределяют участников по типам личности, таким как реалистический, исследовательский, артистический, социальный, предпринимательский и конвенциональный, что помогает уточнить подходящую сферу занятости.

Тесты на оценку профессиональных склонностей и способностей. Эти тесты оценивают уровни развития определённых навыков, таких как аналитическое мышление, внимание к деталям, коммуникативные или лидерские качества, что позволяет определить, какие области могут подходить человеку. Например, тесты на когнитивные способности выявляют силу в таких областях, как математическое или вербальное мышление, что может помочь в выборе профессий, требующих аналитических или языковых навыков.

Методики на оценку личностных качеств. В рамках профориентации также часто используются тесты на личностные качества, такие как 16-факторный личностный опросник Кэттелла или MBTI (Майерс-Бриггс). Эти тесты помогают понять, как человек взаимодействует с окружающим миром, принимает решения, предпочитает ли структурированную деятельность или гибкие задачи. Результаты таких тестов показывают, как черты характера могут способствовать или мешать успешной работе в тех или иных профессиях.

Практические упражнения и кейс-задания. В некоторых профориентационных тестированиях используются упражнения, имитирующие реальные профессиональные задачи. Это может быть решение логических задач для будущих программистов или кейс-задания на общение для тех, кто склонен к работе с людьми. Подобные методы позволяют выявить способности человека в реальных условиях, и такие данные помогают сопоставить его с профессиями, требующими аналогичных навыков.

Эти традиционные методы широко применяются и сегодня, так как дают возможность объективно оценить предпочтения и склонности человека, не прибегая к сложным технологиям. Однако их результаты ограничены формальными рамками и могут не учитывать тонких психологических и социальных факторов, проявляющихся в индивидуальном поведении.

В современных профориентационных системах классические методы могут быть усилены искусственным интеллектом и большими языковыми моделями (LLM), что позволяет более точно анализировать личностные черты, предпочтения и даже скрытые интересы человека. Например, LLM могут не только интерпретировать текстовые ответы, но и создавать связи между личностными характеристиками и особенностями разных профессий, обеспечивая более точные и детализированные результаты профориентации.

Основная цель системы профориентационного тестирования – помочь абитуриентам определить факультет, который наиболее соответствует их личным качествам, интересам и профессиональным наклонностям. Система должна проанализировать ответы на серию вопросов, выявить ключевые черты, сопоставить их с требованиями различных факультетов и предложить наиболее подходящее учебное направление.

**Определение требований к системе**

1. **Точность и обоснованность рекомендаций.** Система должна обеспечивать высокую точность в определении подходящего факультета, основываясь на ответах пользователя. Для этого необходимо использовать методы анализа, которые способны выявлять как явные, так и скрытые признаки, указывающие на склонности и способности абитуриента.
2. **Простота и удобство использования.** Интерфейс должен быть интуитивно понятен, чтобы тестирование было комфортным и легкодоступным для школьников и студентов. Это включает понятную структуру вопросов и простую навигацию по этапам теста.
3. **Интерактивность и визуализация результатов.** Для лучшего понимания результатов тестирования система должна наглядно представлять результаты (например, через графики или схемы), показывающие уровни развития различных профессиональных качеств пользователя и их сопоставление с характеристиками факультетов.
4. **Гибкость анализа ответов.** Платформа должна учитывать текстовые ответы и сопоставлять их с определенными характеристиками на основе LLM, что позволяет учитывать нюансы, характерные для каждого респондента, и получать более персонализированные результаты.

**Ключевые характеристики тестирования**

1. **Структура вопросов. Тест состоит из 25 вопросов, из которых 3** – **текстовые, а остальные** – **с оценкой уровня профессиональных качеств (всего 22 качества). Этот набор вопросов охватывает широкий спектр характеристик, важных для учебной и профессиональной деятельности.**
2. **Анализ ответов с помощью LLM. LLM используется для обработки текстовых ответов и создания уникального профиля пользователя, что повышает точность в определении подходящего факультета.**
3. **Методика расчёта итогового результата. Результаты тестирования представляются в виде вектора, который затем сравнивается с целевыми векторами факультетов с использованием косинусного расстояния. Это позволяет ранжировать факультеты по степени соответствия профилю пользователя.**
4. **Формирование персонализированных рекомендаций. Помимо ранжирования факультетов, система должна предоставлять объяснение, как выявленные качества влияют на соответствие определённому факультету, помогая пользователю понять, почему ему предложен именно этот выбор.**

**Ожидаемый результат**

Разработанная система должна помочь пользователю осознанно выбрать факультет, максимально подходящий к его личностным характеристикам и профессиональным склонностям, а также предоставить прозрачные и понятные обоснования этому выбору.

# 2 ПРОЕКТИРОВАНИЕ

**2.1 Общая концепция системы**

Система профориентационного тестирования предназначена для абитуриентов, чтобы определить наиболее подходящий факультет на основе их ответов на тестовые вопросы. Она представлена как веб-приложение, где серверная и клиентская части выполняют разные, но взаимодополняющие функции.

**Структура системы**

**Клиентская часть** (веб-интерфейс) – предоставляет пользователю доступ к тестированию и визуализации результатов.

**Серверная часть** – обрабатывает данные, анализирует ответы и взаимодействует с локальной языковой моделью (LLM) для формирования персонализированных рекомендаций.

**Распределение задач между серверной и клиентской частями**

**Клиентская часть** отвечает за взаимодействие с пользователем через веб-интерфейс, предоставляя легкий доступ к тесту.

Содержит основные разделы:

* Приветственная секция – объясняет пользователю цель тестирования и общую концепцию.
* Секция с формой – отображает 25 вопросов, из которых 3 требуют произвольных текстовых ответов, а 22 направлены на оценку определенных профессиональных качеств. После завершения ответов пользователь запускает анализ с помощью кнопки.
* Секция результатов – визуализирует итоговый профиль пользователя на графике (звездочная диаграмма) и позволяет переключаться между векторами ответов (тест, LLM и итог).
* Секция статистики – показывает средние результаты тестирования всех участников по каждому критерию.

Обеспечивает передачу ответов на сервер для анализа и отображает полученные результаты. Визуализирует топ-5 факультетов, сортированных по степени соответствия, и объясняет причины выбора факультета, который занял первое место.

**Серверная часть:** обрабатывает ответы пользователя и формирует запросы к LLM для анализа текстовых ответов.Выполняет формирование итогового вектора на основе ответов:

* Текстовые ответы преобразуются в вектор с помощью LLM, который возвращает значения по 22 критериям.
* Ответы на вопросы с выбором интерпретируются и объединяются с вектором, полученным от LLM, для создания итогового вектора, характеризующего пользователя.
* Выполняет сравнение итогового вектора с целевыми векторами каждого факультета с помощью косинусного расстояния. Это позволяет выявить степень соответствия пользователя различным факультетам и сформировать список из пяти наиболее подходящих направлений.
* Генерирует персонализированное объяснение выбора факультета, включая рекомендации на основе индивидуальных качеств, выявленных в процессе тестирования.
* Отправляет результаты обратно на клиентскую часть для отображения пользователю.

**Логика взаимодействия**

1. Пользователь заполняет форму на клиентской части и отправляет её.
2. Ответы направляются на сервер, где выполняется обработка данных и формирование итогового профиля.
3. Сервер передает результаты клиентской части, которая визуализирует их и предоставляет рекомендации пользователю.

Такое распределение задач позволяет оптимально использовать серверные ресурсы для обработки данных и обеспечивает удобный интерфейс для пользователя, делая профориентационное тестирование простым, наглядным и эффективным.

**2.2 Выбор технологий и инструментов**

**Клиентская часть**

Для создания клиентской части проекта выбран **Next.js**, который сочетает удобство разработки React-приложений с возможностями серверного рендеринга. Это облегчает отображение интерфейса, особенно с использованием библиотек, таких как **Ant Design**, которая предоставляет готовые компоненты, соответствующие современным требованиям к UI/UX, что помогает сделать интерфейс удобным и эстетичным.

**Управление состоянием**

Для управления состоянием использован **Redux** с набором инструментов **@reduxjs/toolkit**. Это позволяет эффективно управлять данными теста и промежуточными результатами, делая структуру хранения данных централизованной и предсказуемой.

**Серверная часть и LLM**

### Для выполнения сложного анализа текстовых данных, связанного с интерпретацией ответов пользователей и генерацией рекомендаций, используется библиотека **Ollama**. Она позволяет интегрировать языковую модель (LLM) в систему и обрабатывать текстовые данные для создания качественных профилей пользователей, что улучшает точность рекомендаций и предоставляет более индивидуализированные результаты.

### **Графическое представление данных**

Для визуализации данных о профиле пользователя и его оценках на основе результатов теста выбрана библиотека **Recharts**. Она поддерживает графики различного типа, что позволяет эффективно отображать результаты в виде звездных графиков и других диаграмм, наглядно показывающих индивидуальные качества пользователя.

**Дополнительные зависимости**

Такие пакеты, как **normalize.css** для нормализации стилей, **numjs** для работы с математическими операциями и **TypeScript** для статической типизации, поддерживают удобство разработки, совместимость и качество кода

# 3 РАЗРАБОТКА

**3.1 Алгоритмы обработки данных**

Для работы системы профориентации и подбора факультета на основе ответов пользователей предусмотрена реализация нескольких этапов обработки данных. Они включают формирование векторов на основе ответов, конвертацию текстовых данных, расчет косинусного расстояния между векторами и генерацию объяснительного текста с помощью LLM. Каждый этап имеет уникальные алгоритмы и особенности, которые описаны ниже.

**Формирование векторов ответов**

На этапе формирования векторов система преобразует ответы пользователя в числовой формат, чтобы можно было их обработать в модели. Функция getAnswers принимает массив ответов пользователя, состоящий из строковых массивов, и создает на его основе описание и вектор:

Описание составляется из первых трех ответов, содержащих основную информацию о сильных и слабых сторонах пользователя.

* Вектор создается путем преобразования оставшихся ответов в числовой формат, где каждая оценка нормируется на шкале от 0 до 1.
* Конвертация текстовых данных в вектор

Конвертация текста описания в вектор производится с помощью модели LLM (в данном случае с использованием Ollama и модели qwen2.5:3b). Эта модель оценивает степень развития качеств пользователя по шкале от 0 до 1. На основе текстового описания пользователя формируется вектор, где каждый элемент соответствует конкретному качеству, например, аналитическое мышление или креативность.

Для преобразования текстового описания в вектор используется API /api/llm/text2vec. В результате работы модели получается массив значений для каждого качества, который можно далее использовать для сравнения с другими векторами.

export async function GET(request: Request) {

try {

const { searchParams } = new URL(request.url);

const message = searchParams.get('text');

if (!message) {

return NextResponse.json({ error: 'Parameter "text" is required.' }, { status: 400 });

}

const systemPrompt = `

Создай описание для каждого качества пользователя на основе их текста, используя шкалу от 0 до 1, где каждое число показывает степень развития этого качества.

Ответ должен быть выведен в формате:

Аналитическое мышление: 0.87

Креативность: 0.43

и так далее для всех качеств.

`;

const qualitiesPrompt = qualities.map(q => {

return `${q.name} – ${q.description}`;

}).join("\n");

const userPrompt = `

Вот текст пользователя: "${message}"

А вот качества, для которых нужно создать описание с оценкой:

${qualitiesPrompt}

Ответ пришли в формате: "Аналитическое мышление: 0.87, Креативность: 0.43, ...".

`;

const response = await ollama.chat({

model: MODEL,

messages: [

{ role: 'system', content: systemPrompt },

{ role: 'user', content: userPrompt }

],

});

const vec = parseResponseToArray(response.message.content)

return NextResponse.json({ response: vec });

} catch (error) {

console.error('Error:', error);

return NextResponse.json({ error: 'Failed to generate response' }, { status: 500 });

}

}

Для определения наиболее подходящего факультета используется расчет косинусного расстояния между векторами пользователя и факультетов. Косинусное расстояние рассчитывается с помощью функции cosineDistance, которая принимает два вектора a и b.

Результаты для каждого факультета сохраняются, после чего производится их сортировка в порядке возрастания расстояния (чем меньше расстояние, тем ближе и релевантнее факультет для пользователя).

**Формирование объяснительного текста**

Формирование объяснительного текста для вывода результатов выполняется с помощью модели Ollama. Система генерирует текст на основе наиболее подходящего факультета и текстового описания пользователя, чтобы пояснить, почему именно этот факультет подходит пользователю. Для этого используется API /api/llm/describe, который генерирует текстовый ответ в формате одного абзаца, начиная с фразы "Результаты обусловлены тем, что...".

export async function GET(request: Request) {

try {

const { searchParams } = new URL(request.url);

const top = searchParams.get('top');

const text = searchParams.get('text');

if (!top) {

return NextResponse.json({ error: 'Parameter "top" is required.' }, { status: 400 });

}

if (!text) {

return NextResponse.json({ error: 'Parameter "text" is required.' }, { status: 400 });

}

const systemPrompt = `

Объясни почему именно этот факультет подходит пользователю используя описание.

`;

const userPrompt = `

Вот самый подходящий факультет для пользователя "${top}"

А вот описание пользователя:

${text}

Ответ в формате одного абзаца, начинающегося с "Результаты обусловлены тем, что" и далее идет твой ответ. Ответ должен быть адресован пользователю, обращайся на Вы.

`;

const response = await ollama.chat({

model: MODEL,

messages: [

{ role: 'system', content: systemPrompt },

{ role: 'user', content: userPrompt }

],

});

return NextResponse.json({ response: response.message.content });

} catch (error) {

console.error('Error:', error);

return NextResponse.json({ error: 'Failed to generate response' }, { status: 500 });

}

}

Этот текст формируется с учетом информации о факультете и личных качествах пользователя, что позволяет сделать ответ более персонализированным и понятным.

**3.2 Реализация интерфейса и взаимодействия**

В разделе рассматриваются основные компоненты пользовательского интерфейса системы профориентационного тестирования, их функциональные особенности и взаимодействие с пользователем. Система построена для создания простого, понятного и увлекательного процесса тестирования, ориентированного на помощь пользователю в выборе наиболее подходящего факультета в Брестском государственном техническом университете.

Приветственная секция служит первым этапом взаимодействия пользователя с системой, где он получает краткую информацию о проекте и его возможностях. Дизайн этой секции включает анимацию с фотографиями студентов, создавая эффект присутствия и вовлеченности.

Компонент WelcomeSection использует библиотеку Ant Design для стилизации текста и кнопок. В центре экрана размещена кнопка «Начать», которая визуально привлекает внимание пользователя. При нажатии на кнопку плавная анимация скроллит страницу вниз, переводя пользователя в секцию с формой тестирования.

Код этого компонента организован таким образом, чтобы при использовании функции scrollBy происходил плавный переход, делая интерфейс более дружественным и привлекательным.

Секция с формой представляет основной этап тестирования, где пользователю предлагается ответить на 25 вопросов. Три из этих вопросов – открытые текстовые, а оставшиеся 22 – вопросы выбора, направленные на оценку уровня развития различных профессиональных качеств.

Компонент FormSection организован для динамической генерации вопросов на основе данных из массива questions. В зависимости от типа вопроса используется один из трех подкомпонентов (SelectQuestion, MultiSelectQuestion, TextInputQuestion). Пользовательские ответы сохраняются в локальном хранилище (localStorage), что позволяет сохранять прогресс даже при случайном закрытии страницы.

После завершения опроса пользователь может нажать кнопку «Отправить», запускающую процесс обработки данных. При нажатии происходит несколько этапов:

Формирование векторов ответов. Текстовые ответы преобразуются в векторный формат с использованием LLM, а ответы на вопросы выбора переводятся в числовые значения.

Создание итогового вектора пользователя. Итоговый вектор включает оценку всех 22 качеств, отображаемую на звездчатом графике, который имеет три состояния: вектор тестирования, вектор нейросети и итоговый вектор.

Сравнение итогового вектора с целевыми векторами факультетов. На этом этапе система сортирует факультеты по степени соответствия пользователю, используя косинусное расстояние.

Секция результатов состоит из нескольких блоков, отображающих итоги тестирования:

* График с визуализацией качеств. Пользователь видит свои результаты по всем 22 качествам и может переключаться между графиками тестирования, ИИ и итогового вектора.
* Топ-5 факультетов. Пользователю предлагается список наиболее подходящих факультетов с указанием уровня соответствия. Эта информация может помочь в принятии осознанного решения о выборе факультета.
* Подробное объяснение. В текстовой форме система предоставляет пояснение, почему был выбран конкретный факультет, опираясь на качества, выявленные в процессе тестирования. Текст сгенерирован LLM, что добавляет индивидуальности ответам.

Компонент ResultsSection взаимодействует с состоянием Redux, где хранятся данные тестирования, чтобы обновлять интерфейс по мере обработки результатов.

В этой секции представлена таблица со средними результатами по 22 профессиональным качествам среди всех участников. Для отображения данных используется компонент Table из Ant Design, который предоставляет удобные функции сортировки и фильтрации. Статистическая информация может помочь пользователям оценить свой уровень относительно других участников и лучше понять свои сильные и слабые стороны.

Завершает интерфейс футер, где указаны разработчики и год создания системы.

# 4 ТЕСТИРОВАНИЕ

**4.1 Тестирование**

**WelcomeSection** (рис. 4.1) – это стартовая секция сайта, которая привлекает внимание пользователя с помощью яркого и современного дизайна. На фоне используются динамические элементы, создающие атмосферу погружения. В центре экрана отображается крупный заголовок "Профориентация с AI", что сразу дает понять, что сайт посвящен профессиональной ориентации с применением искусственного интеллекта. Под заголовком идет краткое описание сервиса: "Узнай к каким профессиям у тебя есть предрасположенность с помощью современных технологий искусственного интеллекта". Это объясняет, что с помощью тестирования на основе AI пользователи могут определить, к каким профессиям у них есть склонности.

Кнопка НАЧАТЬ расположена под описанием и стилизована в светлом фоне с синим текстом, чтобы выделяться. При нажатии на эту кнопку страница плавно прокручивается вниз к разделу тестирования, что создает приятный пользовательский опыт.

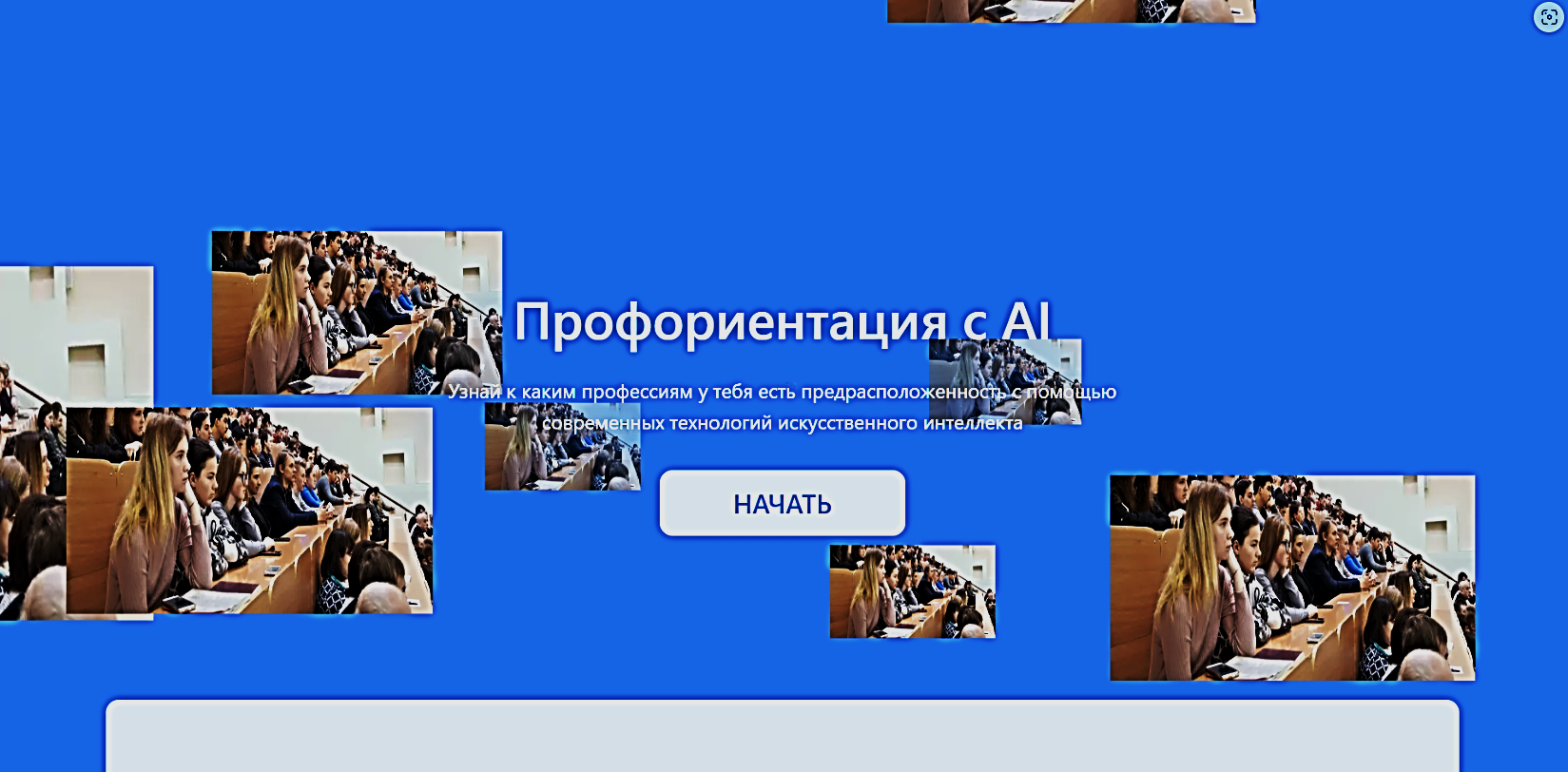


Рисунок 4.1 – WelcomeSection

**FormSection** (рис. 4.2) – это секция сайта, где пользователь отвечает на вопросы теста, чтобы определить свои склонности и подходящие направления для обучения. Она отвечает за отображение вопросов, сбор ответов и вычисление результатов на основе введенных данных.

Загрузка сохраненного состояния: при загрузке страницы проверяется наличие сохраненных ответов в localStorage. Если такие данные есть, они загружаются, чтобы пользователь мог продолжить тест с того места, на котором остановился.

Отображение вопросов: для каждого вопроса вызывается функция getQuestionItem, которая возвращает соответствующий компонент в зависимости от типа вопроса:

SelectQuestion – вопрос с одиночным выбором.

MultiSelectQuestion – вопрос с множественным выбором.

TextInputQuestion – вопрос с текстовым вводом.

Компоненты вопросов получают функции setValue и saveValue для обновления значений в answers и сохранения данных в localStorage.

Сохранение ответов: при изменении ответа вызывается setValueWithIndex, обновляющая состояние answers. Если пользователь завершил ответ на вопрос, saveValueWithIndex сохраняет данные в localStorage, чтобы они не потерялись при обновлении страницы.

Отправка формы: когда пользователь нажимает кнопку "Отправить", выполняется проверка всех ответов с помощью validateTest. Если есть незаполненные вопросы, появляется уведомление об ошибке. В противном случае запускается процесс вычисления результатов.

Отправка данных теста: ответы пользователя обрабатываются с помощью getAnswers, чтобы получить описательный текст и векторное представление (test.description и test.vector).

Косинусная близость: вектор пользователя сравнивается с векторами для различных факультетов с помощью функции cosineDistance, чтобы рассчитать сходство и выбрать наиболее подходящие направления.

Анализ AI: результат передается модели AI через API /api/llm/text2vec, и ответ AI используется для коррекции исходного вектора. На основе этих данных строится итоговый вектор final, который отражает коррекцию от AI.

Анализ факультетов: на основе итогового вектора final для каждого факультета вычисляется средняя дистанция (сходство) для набора студентов, ассоциированных с каждым факультетом. Это позволяет выявить факультеты с наибольшей близостью к пользователю.

Сохранение и отображение результатов: результаты теста сохраняются в Redux, чтобы их можно было использовать в других частях сайта. Визуализация данных и описание выводятся через уведомления и обновляют соответствующие элементы интерфейса:

График результатов: отображается на графике, который наглядно показывает пользователю его результаты по сравнению с рекомендованными направлениями.

Рекомендованный факультет: определяется лучший факультет, а для него через API /api/llm/describe AI создает детальное описание, объясняющее, почему это направление подходит.

Интерактивность: после отправки формы кнопка "Отправить" скрывается, чтобы предотвратить повторные нажатия, а уведомления информируют пользователя о ходе процесса и его завершении.

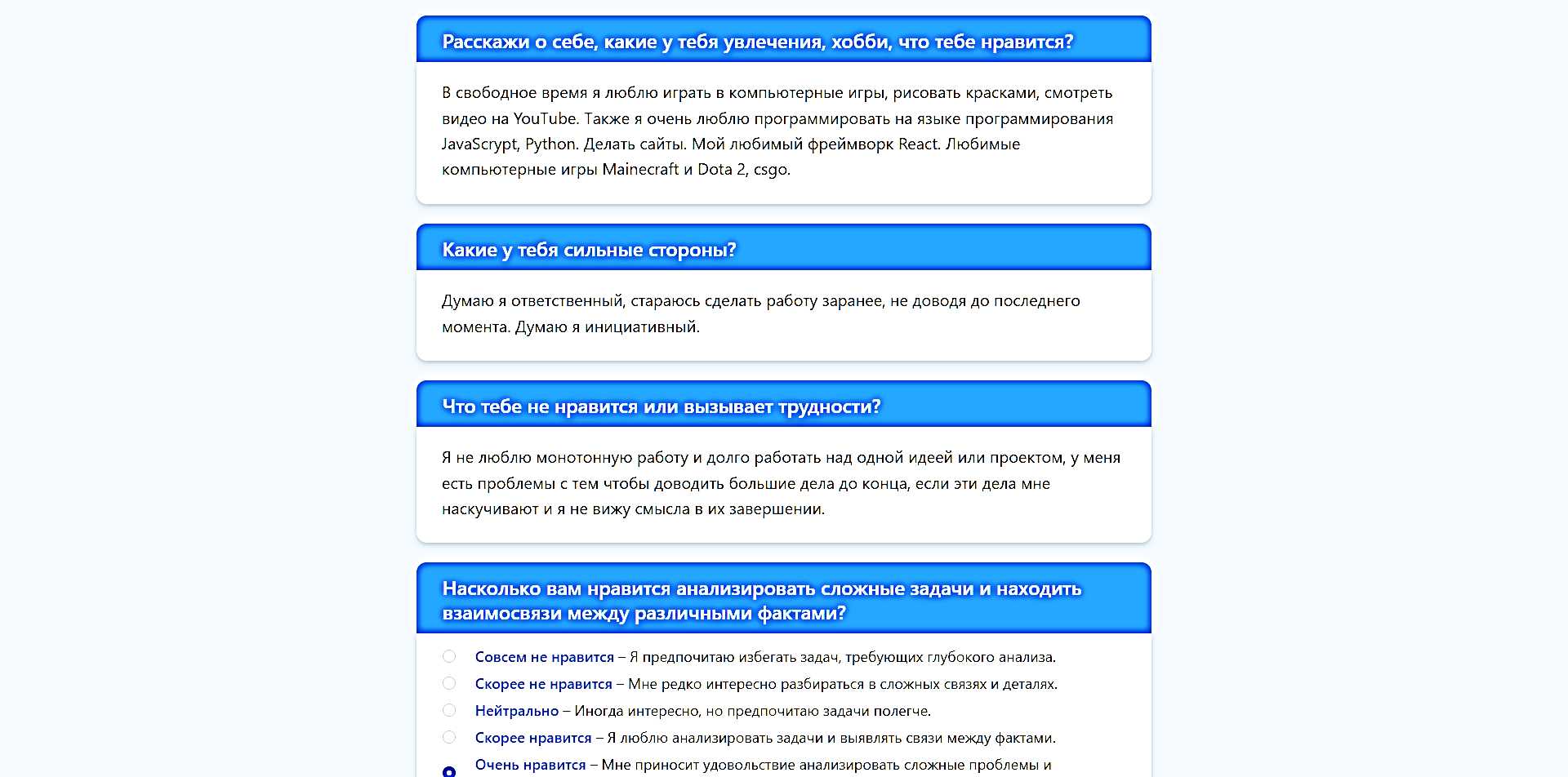
**

Рисунок 4.2 – FormSection

В компоненте **ResultsSection** отображаются результаты тестирования пользователя. Основной заголовок "Ваши результаты тестирования" выделен синим цветом для привлечения внимания. Внутри ResultsSection размещены два основных элемента: Chart и TopList, которые представляют результаты в виде графиков и списка.

Компонент **Chart** (рис. 4.3) – это радиальный график (Radar Chart), который визуализирует сравнительные данные по нескольким качествам пользователя, оцененным в ходе теста. График имеет три слоя данных:

Тест – результаты, полученные непосредственно в процессе тестирования (синие линии и заливка).

AI – данные, обработанные искусственным интеллектом для предсказания способностей пользователя (фиолетовые линии и заливка).

Итог – финальные результаты, комбинирующие данные теста и AI-оценки (зелёные линии и заливка).

Пользователь может переключаться между этими слоями с помощью радиокнопок, расположенных под графиком. При выборе одного из слоёв отображается соответствующая метрика, что позволяет сравнить результаты разных оценок.

Компонент также использует **ChartSkeleton**, отображающий скелетон-загрузчик (placeholder), когда данные ещё загружаются, обеспечивая плавный пользовательский опыт.

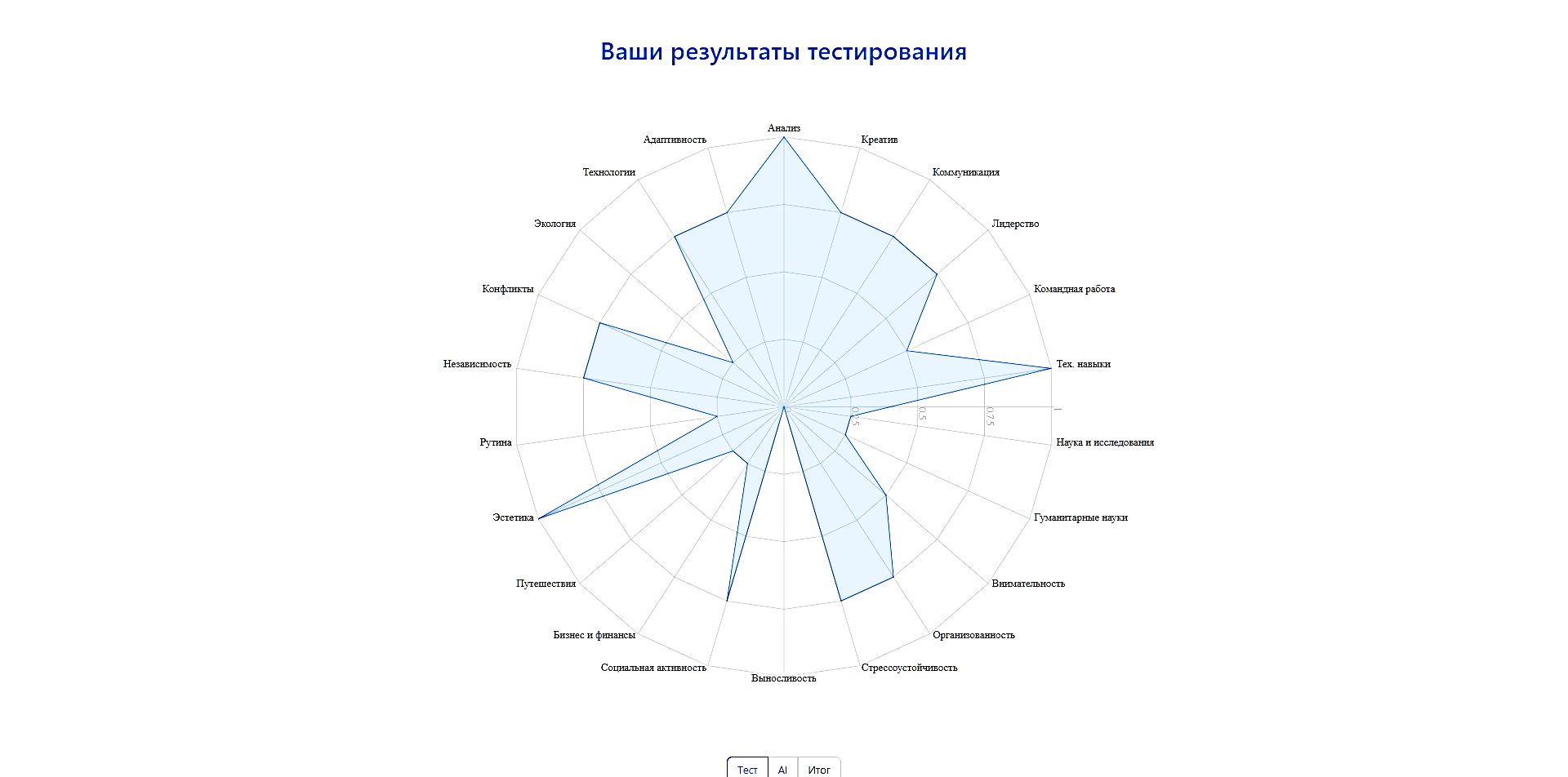
**

Рисунок 4.3 – ResultSection Chart

Компонент **TopList** (рис. 4.4) отображает таблицу с информацией о пяти наиболее подходящих факультетах для пользователя, а также текстовое описание результата, чтобы предоставить дополнительные рекомендации и пояснения.

**Структура таблицы:** Таблица состоит из трех столбцов:

* № – Позиция факультета в рейтинге. Здесь используются эмодзи, чтобы выделить каждый из топ-5 факультетов (🏆, 2️⃣ , 3️⃣ и так далее), добавляя визуальную привлекательность.
* Факультет – Название факультета, подходящего пользователю, отображается крупным и легко читаемым текстом.
* Оценка, % – Рейтинг, рассчитанный на основе результатов теста, показывается в процентах с округлением до одного знака после запятой, что делает данные более наглядными и точными.

Данные: Рейтинг факультетов и их оценки передаются в компонент как массивы faculties и scores. Эти данные преобразуются в формат, удобный для отображения, с округлением оценки и присвоением каждому факультету соответствующего эмодзи-позиции.

Под таблицей находится карточка (Card), в которой выводится текстовое описание результатов (resultDesc). Это описание содержит пояснения и советы, которые помогают пользователю лучше понять свои результаты теста и делают рекомендации более индивидуализированными.

Если данные ещё не загружены, компонент отображает TopListSkeleton, который служит плейсхолдером и показывает схематичное представление таблицы, чтобы обеспечить плавный пользовательский опыт и визуальное подтверждение загрузки информации.

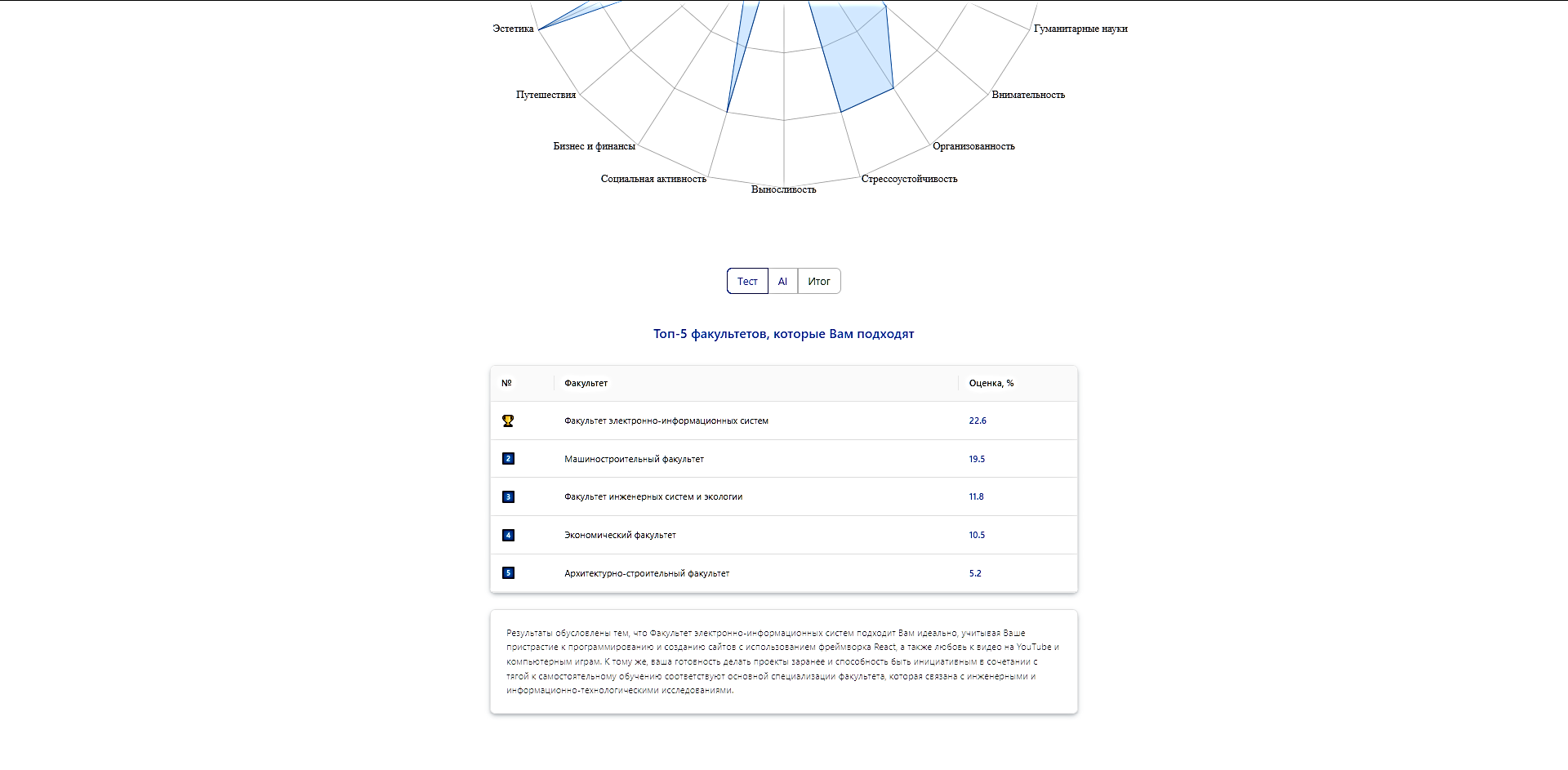
**

Рисунок 4.4 – ResultSection TopList

Компонент **FooterSection** (рис. 4.5) в данном разделе сайта отображает таблицу, которая содержит средние показатели различных качеств пользователей, участвующих в тестировании, а также краткую информацию о проекте в подвале страницы.

**Структура таблицы:** Таблица состоит из трех столбцов:

* Качество – Название личного качества (например, аналитическое мышление, креативность и другие), важного для анализа результатов теста. Названия качеств выделены стилем для улучшения визуального восприятия.
* Описание – Краткое описание каждого качества, поясняющее его суть и важность. Это помогает пользователям лучше понимать, что измеряется и какие качества у них анализируются.
* % – Средний процентный показатель, отражающий степень проявления каждого качества среди всех участников теста. Этот показатель рассчитывается как среднее значение по всем пользователям и округляется до целых процентов для наглядности.

Для каждой строки таблицы используются данные из массива qualities, содержащего названия и описания качеств, и массива students, в котором хранятся показатели пользователей. Процентные значения для каждого качества рассчитываются как среднее среди всех участников, позволяя получить общую картину и сравнить свои результаты с результатами других пользователей.

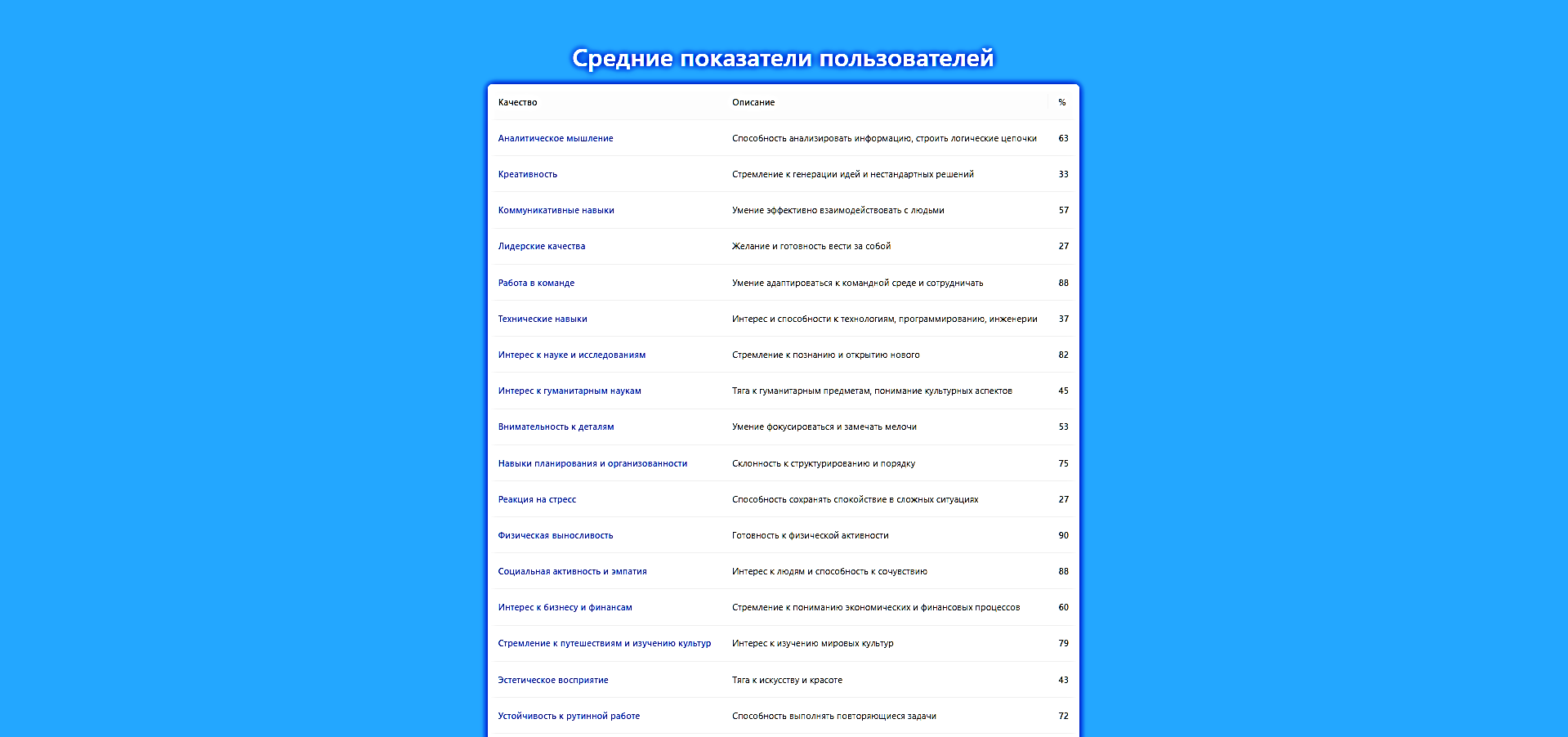


Рисунок 4.5 – FooterSection

По итогам теста система подтвердила то, что факультет электронных интеллектуальных систем наиболее подходящий для меня, что соответствует моему собственному мнению.

# ЗАКЛЮЧЕНИЕ

Разработанная система профориентационного тестирования, основанная на больших языковых моделях (LLM), представляет собой инновационный инструмент для помощи молодым людям в выборе будущей профессии. Система эффективно сочетает в себе психометрические методы оценки с возможностями искусственного интеллекта. Пользовательский интерфейс, интуитивно понятный и визуально привлекательный, обеспечивает комфортное прохождение тестирования и получения результатов.

В основе системы лежит алгоритм, который анализирует ответы пользователя на вопросы теста и сравнивает их с профилями успешных представителей различных профессий. Использование LLM позволяет не только определить наиболее подходящие направления обучения, но и предоставить персонализированные рекомендации, основанные на глубоком понимании текста. Результаты тестирования визуализируются в интерактивных графиках и таблицах, что облегчает их восприятие. Система также предоставляет подробное описание каждого рекомендованного факультета, помогая пользователю сделать осознанный выбор.

Разработанная система демонстрирует высокую эффективность в определении профессиональных склонностей и может стать ценным инструментом для школьников и студентов при выборе будущей карьеры. Дальнейшие исследования могут быть направлены на расширение базы данных профессий, улучшение алгоритмов анализа данных и интеграцию системы с другими образовательными платформами.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Next.js [Электронный ресурс]. – 2024. – Режим доступа: <https://nextjs.org/> – Дата доступа: 08.11.2024.

2. Ollama [Электронный ресурс]. – 2024. – Режим доступа: <https://ollama.com/> – Дата доступа: 08.11.2024.

3. Ant Design [Электронный ресурс]. – 2024. – Режим доступа: <https://ant.design/> – Дата доступа: 08.11.2024.