

VIBECODE JAM

Собеседование
будущего

Создание системы автопроверки



СТРЕСС или СТРАХ?

В российском опросе Netology две трети людей, ищущих работу, отмечали стресс, а 18% прямо боятся проходить собеседования.

НЕ ВОЛНУЙСЯ ОНИ ЧУВСТВУЮТ

По данным мэта-анализа чем выше тревога на интервью, тем ниже оценка интервьюером. Корреляция около 0.19, то есть прослеживается заметная связь

ХОЧЕТСЯ ЛИ ТЕПЕРЬ РАБОТАТЬ?

Российский опрос по стресс-интервью:

- 43% кандидатов называли среди признаков стресс-интервью грубое поведение и провокационные вопросы;
- 32% — перебивания и торопление со стороны интервьюера;
- 29% — навязчивые вопросы о личной жизни.

Треть опрошенных сталкивались с таким

ЭЙ! А ОН МНЕ НРАВИТСЯ

до 90% решений о найме искажены субъективными впечатлениями, не связанными с реальными навыками

В ИТ ПРОБЛЕМНЕЕ

Классические «whiteboard / live-coding»-интервью создают сильную performance anxiety:

- жёсткие лимиты по времени (30–45 минут на часто непростые задачу);
- чужой IDE
- интервьюер сидит и смотрит на каждый символ, что усиливает самокритику и блокировку мыслей.

ИНФОРМАЦИЯ У ВСЕХ

Рекрутеры могут создавать вакансии, откликающиеся на них соискатели проходят интервью, которое соответствует по сложности грейду, а по темам - профилю вакансии. После прохождения собеседования с алгоритмическими задачами и задачами по теории профиля вакансии каждый собеседуемый получает подробный отчет с результатами общения и полезной информацией о возможном развитии. Рекрутер же получает полный профиль кандидата вместе с его положением в рейтинге прошедших интервью.

The screenshot shows two main parts of the VibeCode platform. On the left, there's a resume analysis interface where a PDF file named 'Резюме Вадима Тимакина.pdf' has been uploaded. Below it, a summary of the candidate's profile is shown: SENIOR, DATA, 5 лет опыта, with skills listed as PyTorch, AutoML, Computer Vision, NLP, Airflow. A detailed analysis of the resume is provided with a confidence level of 98%. On the right, a card displays a specific task: 'Как была подобрана задача:' (How was the task selected?). It lists the TPEK as 'MI', the complexity as 'Hard', and the skills required as 'Python, MI, Data Analysis'. It also includes a section titled 'ПОЧЕМУ ИМЕННО ОНА:' (Why specifically this one?) which describes the task 'Merge Intervals' as a test of algorithmic and array skills, relevant for ML engineers.

The screenshot shows the VibeCode registration and login page. At the top, the logo 'VibeCode AI-powered technical interviews' is displayed. Below it, there are two buttons for quick access: 'Войти как кандидат' (Candidate login) in blue and 'Войти как рекрутер' (Recruiter login) in purple. A 'ИЛИ' (OR) button separates these from the main login fields. The main form includes fields for 'Имя' (Name), 'Ваше имя' (Your name), 'Роль' (Role) with options 'Кандидат' (Candidate) and 'Рекрутер' (Recruiter), 'Email' (olegi@olegi.g), 'Пароль' (password), and a 'Зарегистрироваться' (Register) button at the bottom.

СТРЕСС? - ПОДСКАЗКИ
ИНТЕРВЬЮ В ВАКУУМЕ? -
ИНТЕРВЬЮ ОТ ВАКАНСИИ
ШАБЛОННЫЕ ЗАДАЧИ? -
АДАПТИВНЫЙ LLM
ПРОВЕРКА НА COPY/PASTE? -
УМНЫЙ КЛАССИФИКАТОР
НЕПОНЯТНАЯ ОЦЕНКА? -
ОЦЕНКА ТОЛЬКО ИЗ
ПОКАЗАТЕЛЕЙ ПЕРФОРМАНСА
ОТВЕТИЛ - ОТПУСТИЛ? -
РЕАЛЬНАЯ ГЛУБИНА ЗНАНИЙ

Вопрос 3 из 20

СРЕДНИЙ ML Теория #3

Что такое обратное распространение ошибки (backpropagation)?

ВАШ ОТВЕТ:

Ведите ваш ответ здесь...

Пропустить Ответить →

0 отвечено | 2 пропущено

СТАТИСТИКА

ОТВЕЧЕНО:

ПРОПУЩЕНО:

ПРАВИЛЬНЫХ:

ПРАВИЛА

- Вернуться к вопросу нельзя
- Пропуск = 0 баллов
- Время ответа учитывается

Обзор Вакансии Кандидаты + Создать вакансию

Управление вакансиями

| НАЗВАНИЕ | ТРЕК | УРОВЕНЬ | КАНДИДАТОВ | ДЕЙСТВИЯ |
|---------------------------|----------|---------|------------|----------------|
| ML Engineer Junior | ML | JUNIOR | 0 | [Edit, Delete] |
| Backend Developer Middle | Backend | MIDDLE | 0 | [Edit, Delete] |
| Frontend Developer Junior | Frontend | JUNIOR | 0 | [Edit, Delete] |
| DevOps Engineer Middle | DevOps | MIDDLE | 0 | [Edit, Delete] |
| Data Engineer Junior | Data | JUNIOR | 0 | [Edit, Delete] |

Обзор Вакансии Кандидаты

«Подлинное знание — знать
границы собственного
незнания.»

Конфуций



| | | |
|--------------|--------|--|
| Алгоритмы | 60/100 | Знает сложные алгоритмы, но код не завершён. |
| Архитектура | 50/100 | Отсутствует оценка архитектуры. |
| Clean Code | 65/100 | Код читаем, но содержит ошибки. |
| Debugging | 50/100 | Нет информации о отладке. |
| Коммуникация | 70/100 | Хорошо структурирует идеи. |

Рекомендации для роста

- 1 Завершайте код перед отправкой.
- 2 Учтесь использовать подсказки эффективно.
- 3 Развивайте понимание архитектуры.

Архитектура

Мы выбрали для решения

- 1 Frontend - React современный, проверенный
 - 2 Backend - Fast API быстро, удобно
 - 3 Чуткий промпting модели Scibox точно, адаптированно

The screenshot shows a laptop screen with a Microsoft Visual Studio Code (VS Code) window open. The window title is "CandidateDashboard.tsx — t1". The code editor displays the following TypeScript code:

```
function CandidateDashboard() {
    // Документы вакансий
    if (loading) {
        <div className="loading">Загрузка вакансий...</div>
    } else if (vacancies.length === 0) {
        <div className="empty-state">
            <p>Пока нет открытых вакансий</p>
            <p className="hint">Вы можете пройти общее интервью выше</p>
        </div>
    } else {
        <div className="vacancies-grid">
            {vacancies.map((vacancy) => (
                <div key={vacancy.id} className="vacancy-card">
                    <div className="vacancy-header">
                        <h4>{vacancy.title}</h4>
                        <span className="level-badge ${vacancy.level}">{vacancy.level}</span>
                    </div>
                    <div className="vacancy-meta">
                        <span>track</span> {vacancy.track}
                        {vacancy.company_name && (
                            <span className="company">{vacancy.company_name}</span>
                        )}
                    </div>
                    {vacancy.description && (
                        <p className="vacancy-desc">{vacancy.description.slice(0, 150)}...</p>
                    )}
                    <button
                        className="apply-btn"
                        onClick={() => startInterview(vacancy.id)}
                    >
                        Пройти собеседование
                    </button>
                </div>
            ))}
        </div>
    }
}
```

The sidebar on the left shows the project structure:

- frontend
- .vscode
- backend
- deploy
- frontend
- node_modules
- public
- src
 - api
 - client.ts
 - questions.ts
 - components
 - questions
 - ChatPanel.tsx
 - GradeProgressB...
 - HintPanel.tsx
 - IdeEditor.tsx
 - MetricsPanel.tsx
 - SkillRadarChart...
 - TaskView.tsx
 - TrustScoreBadg...
 - pages
 - AdminDashboard...
 - CandidateDashb...
 - InterviewPage.tsx
 - LandingPage.tsx
 - LoginPage.tsx

The bottom of the screen shows the terminal output: "chrizantona@MacBook-Air-alan t1 %".

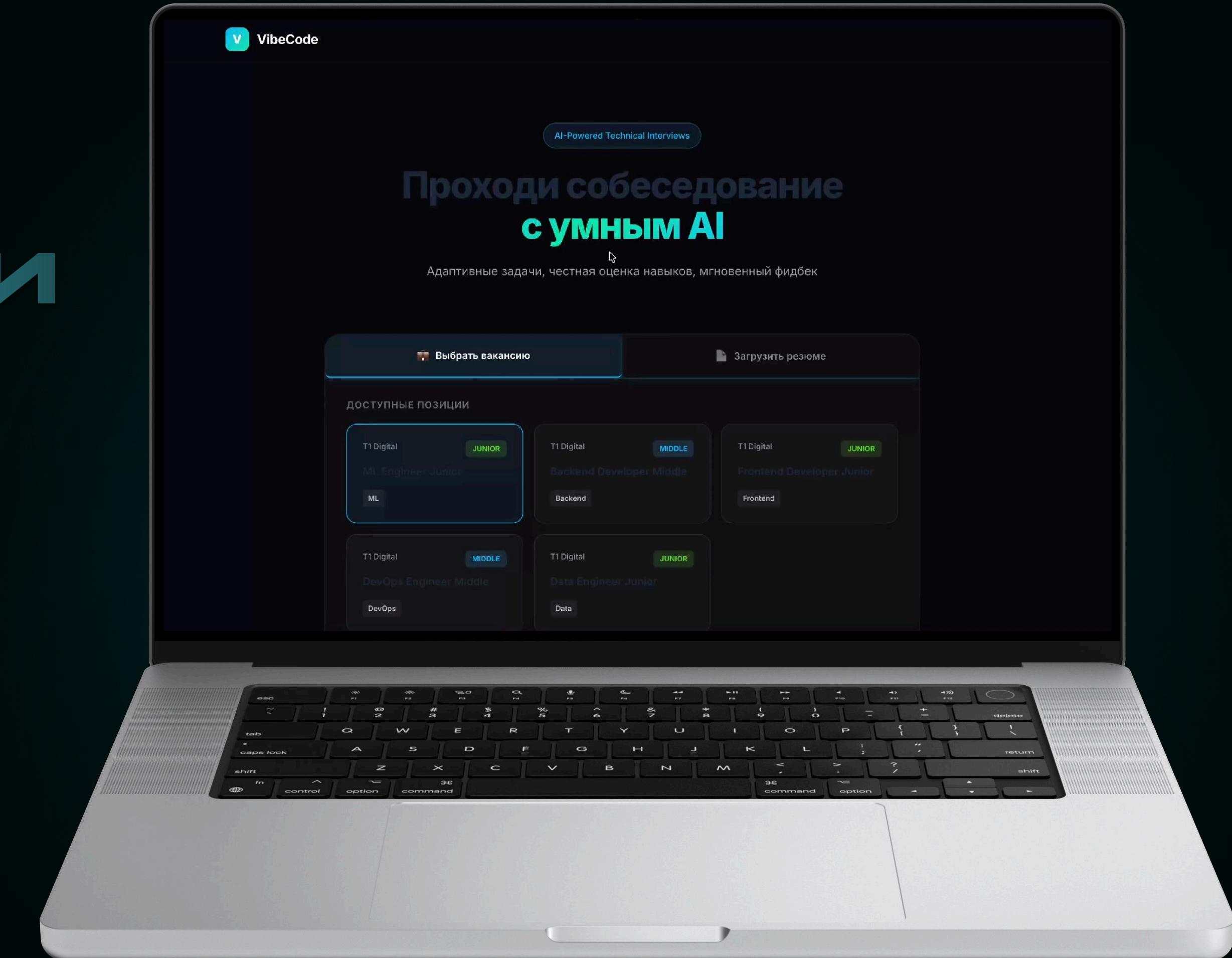
«МАСШТАБИРОВАНИЕ»

ПОЛНОЦЕННЫЙ НАДЕЖНЫЙ ПРОКТОРИНГ С
ВЕБ-КАМЕРОЙ

ГОЛОСОВОЕ ИНТЕРВЬЮ С ИИ

ЛОНГИТЮДНЫЙ ПРОФИЛЬ КАНДИДАТА

**РЕШАЙТЕ САМИ
КТО ЗАДАЕТ
ВОПРОСЫ**



Халибеков Алан



FrontEnd / DevOps

@nnnairobi

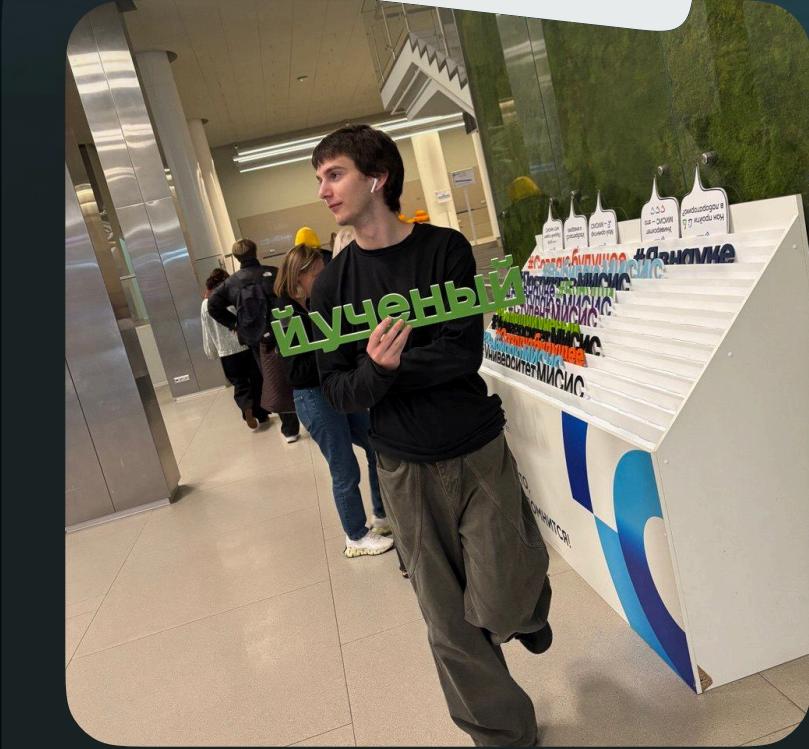
Хамидуллин Тимур



ML

@timjur1

Аракелян Артур



UX/UI designer

@sar_a_to_v

Ананьев Даниил



BackEnd / DevOps

@rougenn

Спасибо за внимание!

Будем рады ответить на ваши вопросы