

Техническое задание: SaaS Клавиатурный Тренажер

1. Описание проекта

Название: Клавиатурный Тренажер (Typing Trainer SaaS)

Версия: 1.0

Дата создания: 2025

Тип проекта: SaaS веб-приложение

1.1 Цель проекта

Разработка современного SaaS-решения для обучения слепой печати на клавиатуре с поддержкой мультиязычности, прогрессивной системой уровней сложности и детальной аналитикой прогресса пользователей.

1.2 Целевая аудитория

- Основная аудитория:**
 - Студенты и школьники (12-25 лет)
 - Офисные сотрудники (25-45 лет)
 - IT-специалисты и программисты (20-40 лет)
- Дополнительная аудитория:**
 - Дети (6-12 лет) - специальный детский режим
 - Преподаватели и учителя информатики
 - Корпоративные клиенты для обучения сотрудников

1.3 Бизнес-модель

Freemium модель:

- Бесплатный базовый функционал
- Premium подписка с расширенными возможностями
- Корпоративные планы для организаций

2. Функциональные требования

2.1 MVP (Минимально жизнеспособный продукт)

2.1.1 Основные функции

- [x] **Тренажер печати**
 - Виртуальная клавиатура с цветовой маркировкой пальцев
 - Подсветка активной клавиши
 - Подсчет скорости печати (WPM - слов в минуту)
 - Отслеживание точности печати
 - Подчеркивание ошибок красной линией
 - Прогресс-бар выполнения
- [x] **Система уровней**

- 6 уровней сложности (Мизинец → Большой палец)
- Тексты разной сложности для каждого уровня
- Цветовая индикация уровней
- [x] **Статистика**
 - Реальная статистика в процессе печати
 - Система рейтинга (1-5 звезд)
 - Сохранение лучших результатов
 - История тестов

2.1.2 Интерфейс

- [x] **Верхняя панель** с мотивирующими цитатами
- [x] **Левая панель** с уровнями сложности
- [x] **Центральная область** - текстовый редактор
- [x] **Правая панель** - статистика в реальном времени
- [x] **Нижняя панель** - виртуальная клавиатура

2.2 Расширенный функционал (v2.0)

2.2.1 Мультиязычность

- [] Поддержка русского и английского языков
- [] Переключение раскладки клавиатуры
- [] Тексты на разных языках
- [] Локализация интерфейса

2.2.2 Детский режим

- [] Упрощенный интерфейс с крупными элементами
- [] Анимированные персонажи-помощники
- [] Игровые элементы и достижения
- [] Звуковые эффекты и награды
- [] Простые тексты и короткие уроки

2.2.3 Персонализация

- [] Регистрация и личные кабинеты
- [] Персональные настройки интерфейса
- [] Темы оформления (светлая/темная)
- [] Настройка сложности

2.3 Премиум функции (v3.0)

2.3.1 Расширенная аналитика

- [] Подробная статистика по буквам и клавишам
- [] Графики прогресса за период
- [] Анализ слабых мест
- [] Рекомендации по улучшению
- [] Экспорт статистики

2.3.2 Дополнительный контент

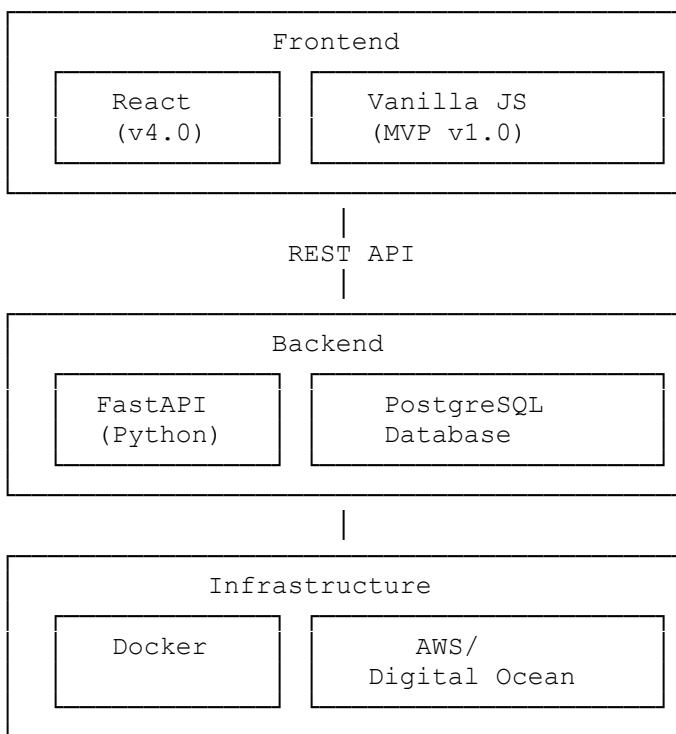
- [] Специализированные тексты (программирование, медицина, право)
 - [] Тематические уроки
 - [] Соревновательный режим
 - [] Мультиплер тренировки

2.3.3 Корпоративные функции

- [] Управление группами пользователей
 - [] Отчеты для преподавателей
 - [] Интеграция с LMS системами
 - [] Белая маркировка

3. Техническая архитектура

3.1 Архитектура системы



3.2 Структура проекта

```
typing-saas/
  └── frontend/
      ├── public/
      └── src/
          ├── components/
          ├── pages/
          ├── hooks/
          ├── services/
          ├── store/
          ├── utils/
          └── styles/
              └── package.json

  └── backend/
      └── app/
```

Фронтенд приложение
Статические файлы
Исходный код
React компоненты
Страницы приложения
Custom hooks
API сервисы
State management
Утилиты
Стили

Бэкенд API
Основное приложение

```

    └── api/          # API роуты
    └── core/         # Конфигурация
    └── models/        # Модели БД
    └── services/      # Бизнес-логика
    └── utils/         # Утилиты
    └── alembic/       # Миграции БД
    └── tests/          # Тесты
    └── requirements.txt

    ├── database/      # База данных
    │   └── migrations/  # Миграции
    │   └── seeds/        # Начальные данные
    └── schema.sql      # Схема БД

    ├── deploy/         # Развертывание
    │   └── docker/       # Docker файлы
    │   └── nginx/        # Конфигурация nginx
    └── scripts/        # Скрипты деплоя

    └── docs/           # Документация
    └── docker-compose.yml  # Локальная разработка

```

3.3 Текущая архитектура (MVP)

Frontend: Vanilla JavaScript + CSS + HTML

- Модульная архитектура с разделением по файлам
- Утилиты для работы с DOM, анимациями, локальным хранилищем
- Компонентный подход к организации кода

Хранение данных: LocalStorage

- Сохранение лучших результатов
- История тестов
- Пользовательские настройки

4. Техстек

4.1 Текущий стек (MVP v1.0)

- **Frontend:** HTML5, CSS3, Vanilla JavaScript ES6+
- **Хранилище:** LocalStorage
- **Инструменты:** Git, VS Code
- **Деплой:** Статический хостинг (Netlify/Vercel)

4.2 Планируемый стек (v2.0+)

Frontend

- **Framework:** React 18+ с TypeScript
- **State Management:** Zustand или Redux Toolkit
- **Стилизация:** Tailwind CSS + Styled Components
- **Сборка:** Vite
- **Тестирование:** Jest + React Testing Library
- **Linting:** ESLint + Prettier

Backend

- **Framework:** FastAPI (Python)
- **Base:** SQLAlchemy ORM + Alembic
- **Authentication:** JWT токены
- **Validation:** Pydantic
- **Тестирование:** pytest
- **Документация:** Swagger/OpenAPI

База данных

- **Основная:** PostgreSQL 14+
- **Кэширование:** Redis
- **Файловое хранилище:** AWS S3 / MinIO

DevOps

- **Контейнеризация:** Docker + Docker Compose
- **Оркестрация:** Kubernetes (production)
- **CI/CD:** GitHub Actions
- **Мониторинг:** Prometheus + Grafana
- **Логирование:** ELK Stack

Инфраструктура

- **Хостинг:** AWS / Digital Ocean
- **CDN:** CloudFlare
- **SSL:** Let's Encrypt
- **Backup:** Автоматические бэкапы БД

5. Дизайн и UX

5.1 Принципы дизайна

- **Минимализм:** Чистый и понятный интерфейс
- **Цветовая система:** Цвета пальцев для обучения
- **Адаптивность:** Поддержка всех типов устройств
- **Доступность:** Соответствие WCAG 2.1

5.2 Цветовая палитра

```
/* Цвета пальцев для клавиатуры */
--pink: #ff7675;          /* Мизинец */
--orange: #fdcb6e;        /* Безымянный */
--green: #00b894;         /* Средний */
--cyan: #74b9ff;          /* Указательный левый */
--blue: #0984e3;          /* Указательный правый */
--purple: #a29bfe;        /* Большой палец */

/* Основные цвета интерфейса */
--primary: #74b9ff;
--success: #00b894;
--warning: #fdcb6e;
--error: #d63031;
```

```
--background: #f5f5f5;  
--surface: #ffffff;  
--text: #2d3436;
```

5.3 Компоненты интерфейса

Верхняя панель

- Мотивирующая цитата дня
- Кнопка запуска теста
- Логотип и навигация

Боковая панель

- Выбор уровня сложности
- Цветовые индикаторы
- Прогресс по уровням

Центральная область

- Текстовый редактор с подсветкой
- Прогресс-бар
- Курсор и анимации

Панель статистики

- WPM в реальном времени
- Процент точности
- Количество ошибок
- Рейтинг звездами

Виртуальная клавиатура

- Цветовая маркировка по пальцам
- Подсветка активной клавиши
- Анимации нажатий

6. База данных

6.1 Схема БД (для v2.0+)

```
-- Пользователи  
CREATE TABLE users (  
    id SERIAL PRIMARY KEY,  
    email VARCHAR(255) UNIQUE NOT NULL,  
    username VARCHAR(100) UNIQUE NOT NULL,  
    password_hash VARCHAR(255) NOT NULL,  
    full_name VARCHAR(255),  
    is_premium BOOLEAN DEFAULT FALSE,  
    is_active BOOLEAN DEFAULT TRUE,  
    created_at TIMESTAMP DEFAULT NOW(),  
    updated_at TIMESTAMP DEFAULT NOW()  
) ;
```

```

-- Тесты
CREATE TABLE tests (
    id SERIAL PRIMARY KEY,
    user_id INTEGER REFERENCES users(id),
    level VARCHAR(50) NOT NULL,
    language VARCHAR(10) NOT NULL,
    text_content TEXT NOT NULL,
    wpm INTEGER NOT NULL,
    accuracy DECIMAL(5,2) NOT NULL,
    errors INTEGER NOT NULL,
    duration INTEGER NOT NULL, -- миллисекунды
    total_chars INTEGER NOT NULL,
    created_at TIMESTAMP DEFAULT NOW()
);

-- Тексты для тренировок
CREATE TABLE training_texts (
    id SERIAL PRIMARY KEY,
    level VARCHAR(50) NOT NULL,
    language VARCHAR(10) NOT NULL,
    title VARCHAR(255),
    content TEXT NOT NULL,
    category VARCHAR(100),
    difficulty INTEGER DEFAULT 1,
    is_active BOOLEAN DEFAULT TRUE,
    created_at TIMESTAMP DEFAULT NOW()
);

-- Достижения
CREATE TABLE achievements (
    id SERIAL PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    description TEXT,
    icon VARCHAR(255),
    criteria JSONB NOT NULL,
    points INTEGER DEFAULT 0,
    is_active BOOLEAN DEFAULT TRUE
);

-- Пользовательские достижения
CREATE TABLE user_achievements (
    id SERIAL PRIMARY KEY,
    user_id INTEGER REFERENCES users(id),
    achievement_id INTEGER REFERENCES achievements(id),
    earned_at TIMESTAMP DEFAULT NOW(),
    UNIQUE(user_id, achievement_id)
);

```

7. API Endpoints (v2.0+)

7.1 Аутентификация

POST /api/auth/register	# Регистрация
POST /api/auth/login	# Вход
POST /api/auth/logout	# Выход
POST /api/auth/refresh	# Обновление токена
POST /api/auth/reset	# Сброс пароля

7.2 Пользователи

GET /api/users/me	# Профиль пользователя
-------------------	------------------------

```
PUT /api/users/me          # Обновление профиля  
GET /api/users/stats       # Статистика пользователя  
DELETE /api/users/me        # Удаление аккаунта
```

7.3 Тесты

```
POST /api/tests           # Сохранение результата теста  
GET /api/tests            # История тестов  
GET /api/tests/{id}         # Конкретный тест  
GET /api/tests/stats        # Статистика тестов  
DELETE /api/tests/{id}        # Удаление теста
```

7.4 Контент

```
GET /api/texts             # Получение текстов для тренировки  
GET /api/texts/{level}      # Тексты по уровню  
GET /api/quotes            # Цитаты для мотивации
```

8. Безопасность

8.1 Аутентификация и авторизация

- JWT токены с ротацией
- Хэширование паролей (bcrypt)
- Rate limiting для API
- CORS настройки

8.2 Защита данных

- HTTPS everywhere
- Валидация всех входных данных
- Защита от SQL инъекций
- XSS защита
- CSRF токены

8.3 Приватность

- Соответствие GDPR
- Анонимизация данных
- Возможность экспорта/удаления данных
- Прозрачная политика конфиденциальности

9. Производительность

9.1 Frontend

- Code splitting и lazy loading
- Оптимизация изображений
- Кэширование статических ресурсов
- Service Workers для офлайн работы

9.2 Backend

- Кэширование запросов (Redis)
- Оптимизация SQL запросов
- Connection pooling
- Асинхронная обработка

9.3 Инфраструктура

- CDN для статических файлов
- Load balancing
- Автоскейлинг
- Database replication

10. Монетизация

10.1 Тарифные планы

Бесплатный план

- Базовый тренажер
- 3 уровня сложности
- Ограниченная статистика
- Реклама

Premium план (\$9.99/месяц)

- Все уровни сложности
- Детальная аналитика
- Персональные настройки
- Без рекламы
- Экспорт данных

Professional план (\$19.99/месяц)

- Все функции Premium
- Специализированные тексты
- Соревновательный режим
- Приоритетная поддержка

Enterprise план (от \$99/месяц)

- Все функции Professional
- Управление командой
- Интеграции
- Белая маркировка
- Персональный менеджер

11. Этапы разработки

11.1 Этап 1: MVP (Завершен)

Сроки: 2-3 недели

Статус: Выполнено

- [x] Базовый интерфейс тренажера
- [x] Виртуальная клавиатура
- [x] Система уровней
- [x] Базовая статистика
- [x] Локальное сохранение данных

11.2 Этап 2: Backend и Auth

Сроки: 3-4 недели

Статус: Планируется

- [] Настройка backend на FastAPI
- [] Создание базы данных
- [] Система аутентификации
- [] API для основных операций
- [] Миграция с LocalStorage

11.3 Этап 3: Расширенный функционал

Сроки: 4-5 недель

Статус: Планируется

- [] Мультиязычность
- [] Детский режим
- [] Улучшенная аналитика
- [] Система достижений
- [] Социальные функции

11.4 Этап 4: Premium функции

Сроки: 3-4 недели

Статус: Планируется

- [] Платежная система
- [] Подписки и тарифы
- [] Корпоративные функции
- [] Интеграции
- [] Мобильное приложение

11.5 Этап 5: Масштабирование

Сроки: 2-3 недели

Статус: Планируется

- [] Оптимизация производительности
- [] Развёртывание в production
- [] Мониторинг и метрики
- [] Автоматическое тестирование

- [] DevOps процессы

12. Маркетинг и продвижение

12.1 Каналы продвижения

- **SEO:** Органический поиск по запросам "тренажер печати", "слепая печать"
- **Контент-маркетинг:** Блог с советами по быстрой печати
- **Социальные сети:** YouTube с обучающими видео
- **Партнерства:** Интеграция с образовательными платформами
- **Реферальная программа:** Бонусы за приведение друзей

12.2 Ключевые метрики

- **MAU** (Monthly Active Users)
- **Retention Rate** (удержание пользователей)
- **Conversion Rate** (конверсия в Premium)
- **ARPU** (средний доход на пользователя)
- **Churn Rate** (отток пользователей)

13. Риски и митигация

13.1 Технические риски

- **Производительность:** Оптимизация кода и использование CDN
- **Безопасность:** Regular security audits и обновления
- **Масштабирование:** Cloud-native архитектура

13.2 Бизнес риски

- **Конкуренция:** Фокус на уникальных функциях и UX
- **Монетизация:** A/B тестирование тарифов
- **Юридические:** Соответствие международным стандартам

14. Поддержка и развитие

14.1 Техническая поддержка

- База знаний и FAQ
- Система тикетов
- Онлайн чат для Premium пользователей
- Email поддержка

14.2 Развитие продукта

- Регулярные обновления функционала
- Сбор обратной связи от пользователей
- Анализ метрик использования
- Roadmap на 12 месяцев

Заключение

Данное техническое задание описывает создание современного SaaS решения для обучения печати на клавиатуре. Проект имеет четкую поэтапную структуру развития от MVP до полнофункциональной платформы с монетизацией.

Ключевые преимущества проекта:

- Современный подход к обучению печати
- Масштабируемая архитектура
- Множественные источники дохода
- Сильный UX и геймификация

Следующие шаги:

- Завершение текущего MVP
- Планирование backend разработки
- Подготовка к привлечению пользователей
- Тестирование монетизации

Версия документа: 1.0

Дата последнего обновления: 25 сентября 2025