

Проектирование онлайн тренажера

для обучения пользователей скринридеров
работе с веб-интерфейсом

Довыденков В. Н.,
группа Р4150

Цель и задачи

- Цель: разработка документации ПО для обучения пользователей скринридеров
- Задачи: анализ, определение требований, разработка архитектуры

Актуальность проекта

- Различия между веб-интерфейсами и нативными интерфейсами
- Затруднения у незрячих пользователей
- Необходимость специализированного тренажера

Опрос пользователей

- Проведен опрос среди целевой группы
- 17 участников, 95% поддержали идею
- Проблемы в веб-интерфейсах: неудобство, непоследовательность

Сбор и анализ требований

- Анализ анкет, формулирование требований
- Навигация, формы, сложные элементы, подсказки
- Поддержка популярных скринридеров и браузеров

Навигация по интерфейсу

The logo for ІТМО (ITMO University) is displayed in white text on a blue rectangular background. The letters are in a bold, sans-serif font. The background of the slide features a dark blue gradient with a lighter blue geometric shape on the right side.

ІТМО

- Сценарии работы с заголовками, ссылками и кнопками
- ARIA-метки для лучшего озвучивания
- Постепенное обучение с практическими заданиями

Работа с формами

- Текстовые поля, чекбоксы, переключатели и др.
- Сложности: управление, озвучивание, навигация
- Решения: инструкции, пошаговые задания

Сложные элементы интерфейса

The logo for IITMO, consisting of the letters 'IITMO' in a white, bold, sans-serif font, set against a bright blue rectangular background.

- Таблицы, списки, модальные окна
- Навигация по вложенным структурам
- Фокусировка, последовательность действий

Встроенные подсказки и справка

- ARIA-описания, tooltips, контекстная справка
- Проблемы: скрытность, нестандартность
- Решения: четкие инструкции и задания

Анализ аналогов

- Отсутствие аналогичных решений
- Существующие инструменты — не тренажёры
- Идея уникальна и актуальна

Нормативные документы

- WCAG 2.2: уровни А, АА, ААА
- ГОСТ Р 52872-2019
- Рекомендации по применению стандартов

Базовые функциональные требования

- Навигация, формы, подсказки — ключевые модули
- Идентификация, сохранение прогресса
- Панель преподавателя

Нефункциональные требования

- Скорость отклика < 3 с
- Совместимость с основными скринридерами и браузерами
- Резервное копирование, безопасность

Архитектура системы



- Клиент: SPA
- Сервер: Flask + SQLAlchemy
- База данных: PostgreSQL

Диаграммы и сценарии

- Диаграммы: последовательности, связи, ER-модель
- Сценарии: ученик, преподаватель
- API-интерфейсы

Выбор технологий

- Flask, React/Vue, PostgreSQL, Docker
- ORM: SQLAlchemy
- Контейнеризация и мониторинг

Заключение

- Решение обосновано аналитикой и опросами
- Разработана структура, сценарии, архитектура
- Есть потребность и нет аналогов

Спасибо
за внимание!

Довыденков В. Н.
+7 9623451818
dov@mlbox.ru