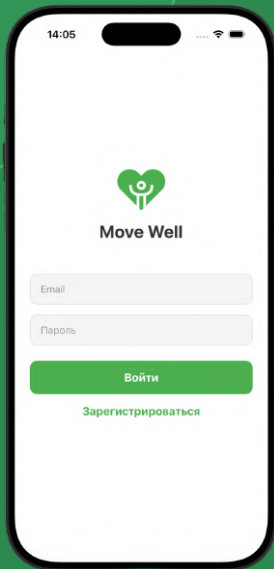


АЛТАЙСКИЙ
ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ

ИМиИТ · Кафедра информатики



Шпигальских
Анастасия Витальевна

студентка группы 4.207-1, 4 курс

MOVE WELL

*Мобильное приложение для реабилитации
людей с ОВЗ и маломобильных граждан*

React Native

TypeScript

iOS & Android

ЧАСТЬ 1

Frontend-разработка мобильного клиента

*Анализ предметной области · UI/UX
Компоненты · REST API · Тестирование*

Науч. рук.: Сливный Д.И.

ЦЕЛЬ ПРОЕКТА



Разработка мобильного приложения «Move Well»

которое на основе технологий компьютерного зрения и персонализированных видео-программ повышает доступность, эффективность и безопасность физической реабилитации для маломобильных людей и лиц с ОВЗ, выступая в роли цифрового ассистента для выполнения предписанных лечащим врачом упражнений.



Доступность

Реабилитация через обычную камеру смартфона — без дорогого оборудования и визитов в клинику



Эффективность

Контроль правильности движений в реальном времени с помощью компьютерного зрения OpenPose



Безопасность

Персонализированные программы с учётом диагноза, уровня подвижности и противопоказаний

АКТУАЛЬНОСТЬ

Почему это важно?



2.4 млн

*инвалидов в РФ
(нарушения ОДА)*



70%

*пациентов бросают
занятия без контроля*



150–3000₽

*в месяц стоит
обычное приложение*



Нет обратной связи

*Фитнес-приложения не проверяют правильность движений —
риск повторных травм*



Недоступность врача

*Регулярные сеансы с физиотерапевтом дороги и требуют
физического присутствия*



Нет персонализации

*Готовые программы не учитывают диагноз, уровень
подготовки и противопоказания*



Дорогое оборудование

*Профессиональное оборудование для анализа движений
недоступно большинству*

АНАЛИЗ СУЩЕСТВУЮЩИХ РЕШЕНИЙ

Функция	Kaia Health	PhysiApp	Fusionetics	Move Well ✦
Анализ движений в реальном времени	Частично	✗	✓	✓
Бесплатный доступ	✗	✗	✗	✓
Адаптация для ОВЗ	✗	Частично	✗	✓
Персональные рекомендации	✓	Частично	✓	✓
Кроссплатформенность	✓	✓	✗	✓
Обратная связь (подсказки)	Частично	✗	✓	✓



Move Well бесплатное решение с детальным анализом движений, адаптированное для людей с ОВЗ

ЗАДАЧИ И ТЕХНОЛОГИИ

Задачи проекта

- 1 Провести анализ предметной области «Цифровая медицина» и рынка.
- 2 Исследовать метода скелетного тренинга (OpenPose) и подходы к сравнению позы
- 3 Спроектировать архитектуру программно-информационной системы
- 4 Провести модульное, интеграционное и юзабилити-тестирование
- 5 Сформулировать рекомендации по внедрению в реабилитационных центрах и для домашнего использования

Стек технологий

 **React Native 0.82**
мобильный фреймворк

 **TypeScript**
строгая типизация

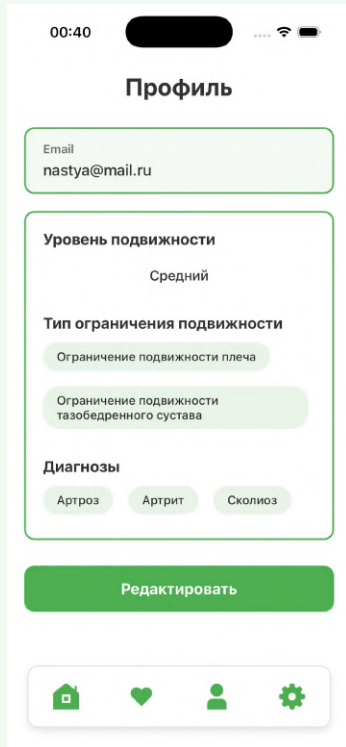
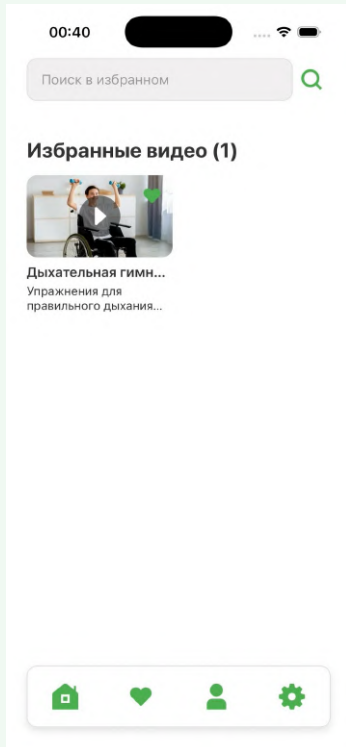
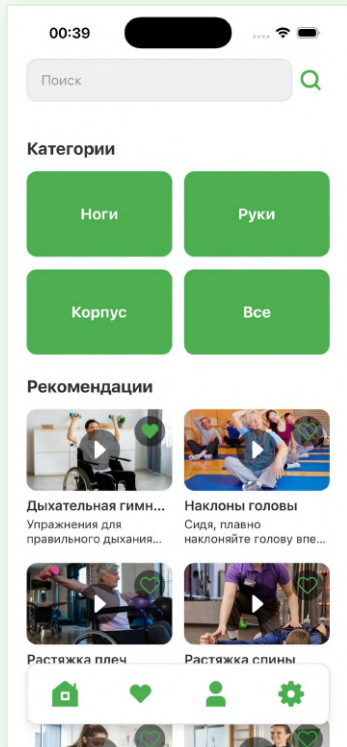
 **Vision Camera 5**
доступ к камере

 **React Navigation**
навигация экранов

 **REST API**
интеграция с сервером

МОБИЛЬНОЕ ПРИЛОЖЕНИЕ

React Native · TypeScript · iOS & Android



Авторизация

Регистрация, вход, JWT



Каталог

13+ экранов, категории, поиск



Избранное

Сохранение упражнений



Профиль

Диагнозы, ограничения, тема



Сессия

Эталонное видео + камера



Темы

Светлая / тёмная

ТЕСТИРОВАНИЕ И РЕЗУЛЬТАТЫ

Кроссплатформенное тестирование

Платформа	Устройство	ОС	Статус
iOS	iPhone 16 Sim	iOS 26	✅ Стабильно
Android	Honor 20	Android 10	✅ Стабильно

Пользовательские сценарии

13

сценариев
успешно
пройдено

на обеих платформах

Достигнутые результаты



Интуитивный интерфейс
13+ экранов



Полная интеграция
с бэкендом (REST API)



Кроссплатформенность
iOS и Android



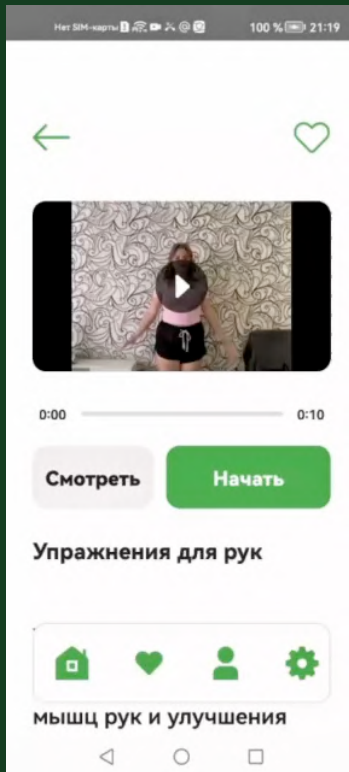
Архитектура с внедрением OpenPose



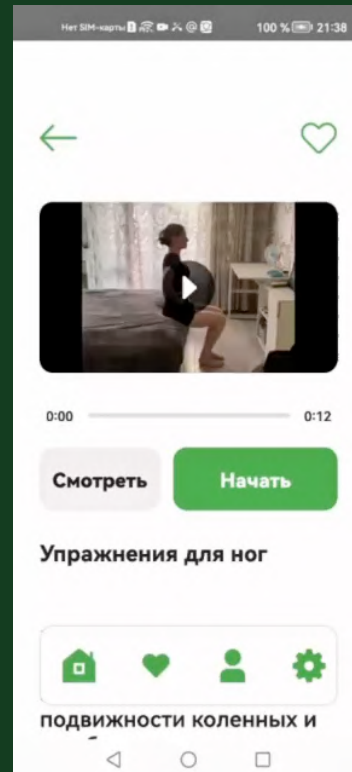
Поддержка светлой
и тёмной темы

ДЕМОНСТРАЦИЯ РАБОТЫ

▶ Демонстрация 1



▶ Демонстрация 2



ЗАКЛЮЧЕНИЕ — ЧАСТЬ 1

Frontend · React Native



Интерфейс реализован

Создан интуитивно понятный пользовательский интерфейс с 13+ экранами: авторизация, каталог, избранное, профиль, настройки, сессия упражнений



Интеграция с бэкендом

Реализована полная интеграция с сервером через REST API: регистрация, авторизация, каталог упражнений, избранное, профиль пользователя



Кроссплатформенность

Приложение протестировано и стабильно работает на iOS (iPhone 16, iOS 26) и Android (Honor 20, Android 10). Все 13 сценариев пройдены



Архитектура для OpenPose

Спроектирована модульная архитектура с компонентом ExerciseSession.tsx и интегрирован модуль компьютерного зрения OpenPose

Все поставленные задачи Frontend-части выполнены в полном объёме

MOVE WELL

Мобильное приложение для
реабилитации людей с ОВЗ

ЧАСТЬ 2

Backend · База данных Компьютерное зрение

FastAPI

PostgreSQL

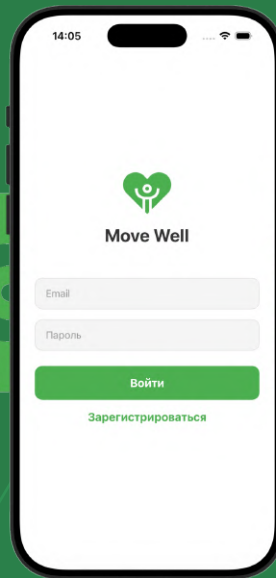
OpenPose

Python 3.11 · SQLAlchemy 2.0 · JWT · WebSocket · OpenCV DNN

Науч. рук.: Сливный Д.И.

АЛТАЙСКИЙ
ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ

ИМИИТ · Кафедра информатики



Ожогина
Татьяна Андреевна

студентка группы 4.207-1, 4 курс

АРХИТЕКТУРА СЕРВЕРНОЙ ЧАСТИ

FastAPI · Python 3.11 · PostgreSQL 15



Мобильный клиент
(React Native)



REST API
(FastAPI)



Бизнес-логика
(Services)



База данных
(PostgreSQL)



auth.py

/register, /login
JWT + pbkdf2_sha256



profile.py

GET/PUT/DELETE
/profile



exercises.py

/exercises + фильтры
/analyze (видео)

services/



pose_analyzer.py

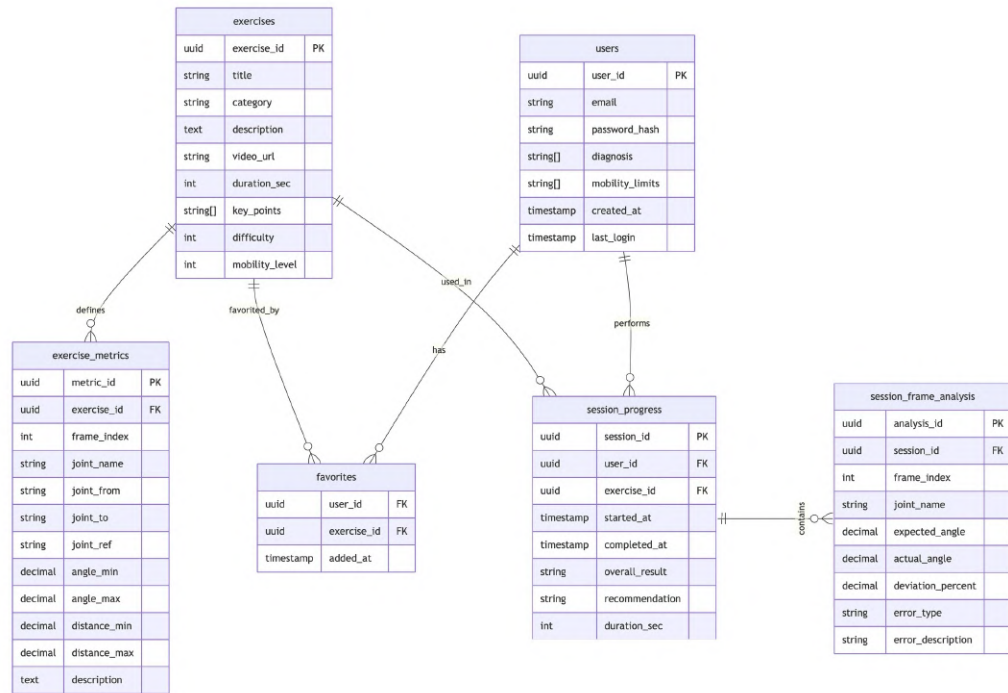
OpenPose + OpenCV DNN
18 ключевых точек



video_analyzer.py

ROM-сравнение
амплитуды движений

БАЗА ДАННЫХ POSTGRESQL



6 таблиц



users

Учётные данные, хеш пароля



user_profiles

Диагнозы, ограничения, тема



exercises

Каталог + эталонные углы (JSONB)



favorites

Пользователь-упражнение (M2M)



session_progress

История сессий, оценка



session_frame_analysis

Потреймовые данные ошибок

REST API

24 тест-кейса · все пройдены 

POST

/auth/register

Регистрация пользователя

POST

/auth/login

Вход, JWT-токен

GET

/profile

Получение профиля

PUT

/profile

Обновление профиля

DELETE

/profile

Удаление аккаунта

GET

/exercises

Список с фильтрацией

GET

/exercises/recommended

Персональные рекомендации

POST

/exercises/{id}/analyze

Анализ видео упражнения

POST

/favorites/{id}

Добавить в избранное

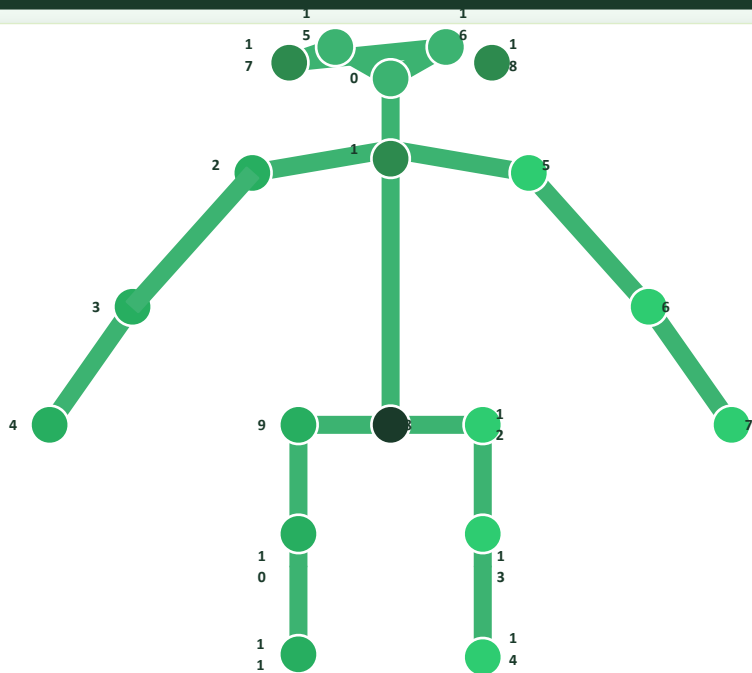
DELETE

/favorites/{id}

Убрать из избранного

МОДУЛЬ КОМПЬЮТЕРНОГО ЗРЕНИЯ

OpenPose COCO · 18 ключевых точек



1

Извлечение кадров
3 кадра/сек

2

Детекция скелета
18 ключевых точек

3

Вычисление углов
8 суставов

4

ROM-сравнение
с эталоном

5

Оценка 0–100
список ошибок

Контролируемые суставы:



Правый
локоть



Левый локоть



Правое плечо



Левое плечо



Правое колено



Левое колено



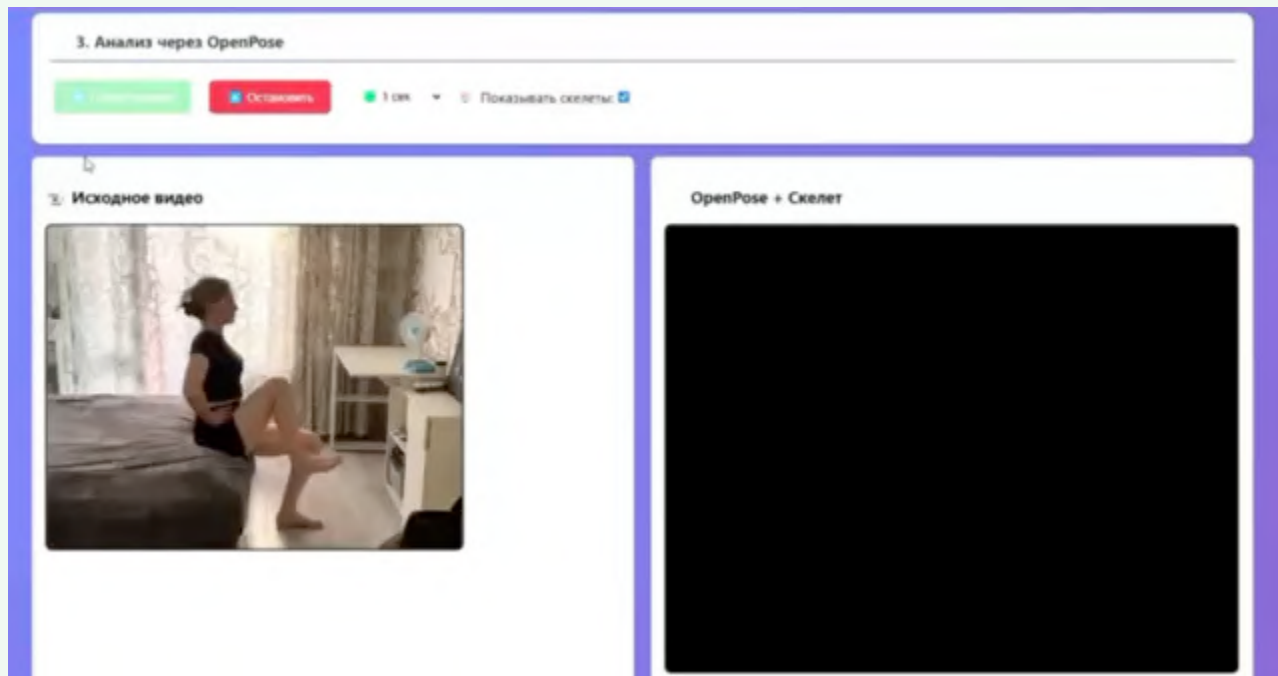
Правый
тазобедренн
ый



Левый
тазобедренн
ый

ДЕМОНСТРАЦИЯ РАБОТЫ OPENPOSE

Тестовое видео



РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ OPENPOSE

Точность детекции PCKh@0.2

Сустав	Точность
Плечи	96%
Бёдра	95%
Колены	94%
Локти	93%
Запястья	89%
Стопы	85%

86.7%

Accuracy

классификация движений

93.3%

Recall

правильные повторения

90.3%

Precision

точность определения



Производительность:

API < 120 мс (p95) · WebSocket 65–105 мс · Полный цикл < 160 мс

ЗАКЛЮЧЕНИЕ — ЧАСТЬ 2

Backend · OpenPose



База данных спроектирована

Реализована схема PostgreSQL из 6 таблиц (users, user_profiles, exercises, favorites, session_progress, session_frame_analysis)



REST API разработан и протестирован

Разработано REST API на FastAPI с полной Swagger-документацией. Проведено функциональное тестирование: 24 тест-кейса — все успешно пройдены



OpenPose исследован и протестирован

Модуль анализа позы на базе OpenPose COCO (18 точек, OpenCV DNN) протестирован на видео. Точность accuracy 86.7%, recall 93.3%, задержка API < 120 мс



ROM-алгоритм реализован

Разработан алгоритм сравнения амплитуды движений (Range of Motion), устойчивый к ракурсу и расстоянию до камеры, с нормализацией по плечевому поясу

Все поставленные задачи Backend-части выполнены в полном объёме

ИТОГИ РАБОТЫ



Мобильный клиент

13+ экранов, React Native, кроссплатформенность iOS & Android



Серверная часть

FastAPI + PostgreSQL, 9 эндпоинтов, 24 тест-кейса



Компьютерное зрение

OpenPose, 18 точек скелета, точность 85-96%



Интеграция

REST API, WebSocket, JWT-авторизация

Спасибо за внимание!