# Pose Correction System for Physical Therapy & Rehabilitation

### What you'll find in the repository

Module	Highlights
Data prep & augmentation	Jupyter notebook named 'model_training_evaluation_and_Inference.ipynb' that clean the original REHAB24-6 corpus and generate augmented training windows.
PoseQualityNet-KP	A 3-head CNN + Bi-LSTM that predicts exercise ID, repetition quality, and 14 joint-angle errors—only 3.4 M parameters.
Training • Evaluation • Ablation	Jupyter Notebook named 'model_training_evaluation_and_Inference.ipynb' that contains the code to reproduce all reported metrics and architecture variants.
RehabApp	Deployable application — FastAPI back-end, React front-end, SQLite persistence, Docker-compose one-liner.
Docs & assets	Techincal Report, Presentation and step-by-step usage guides.

**RehabApp** is an end-to-end Al physiotherapy coach that turns an ordinary webcam into a real-time form-correction assistant.

This repository bundles everything you need—from data preparation to a deployable FastAPI + React stack.

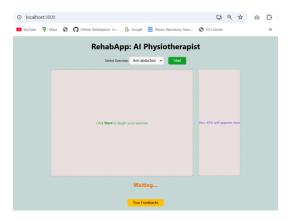
## Repository Layout

Path	What's inside
Data/	Contains links and information about the data sources used.
└ Data Source Links.txt	Text file with links/references to the original and processed datasets.
Data Preprocessing - Model Creation and Evaluation/	Data-prep helpers and the main Jupyter notebook
—model_training_evaluation_and_Inference.ipynb	Data augmentation, model creation, evaluation, ablation study & inference
└─ Data-REHAB24-6_New/	Augmented window tensors & meta-data generated by the notebook

Path	What's inside
Documentation/	Project documentation, reports, and guides.
└Presentation/	Slides (PDF) and recorded project presentations.
└─ Project Report/	Detailed project report (PDF and Latex code)
└─ User Guide/	Instructions and guides for using Repo and RehabApp.
rehab-app-backend/	FastAPI service (main.py, model loader, SQLite storage, Dockerfile)
rehab-app-frontend/	React client (pose capture, WebSocket, dashboard UI)
docker-compose.yml	One-shot build & launch for both apps

### **♦** Running the RehabApp Application:

- 1. Clone the repository:
  - o git clone <a href="https://github.com/IS06-Jithin/RehabApp.git">https://github.com/IS06-Jithin/RehabApp.git</a>
- 2. Command prompt/terminal navigate to RehabApp folder
  - o cd RehabApp
- 3. Build and run with Docker Compose:
  - o docker-compose up --build
- 4. Access the application: Once the build is complete and containers are running:
  - React Client (Frontend): <a href="http://localhost:3000/">http://localhost:3000/</a> (allow webcam access when prompted)



#### 5. Using the App

- o Select the exercise from the dropdown menu and click the **start** button
- Perform the selected Exercise
- Pay attention to the visual/Auditory feedback and correct yourself
- Upon completion, click the **Stop** button and review your session summary
- If you have any feedback on the App's usability, click the feedback button and provide it.

## **X** Prerequisites

- Docker & Docker Compose:
  - o Windows / macOS: Install Docker Desktop from docker.com.
  - Linux: Install the Docker Engine and the docker-compose plugin from your distribution's package manager or follow the <u>official guide</u>.
- **Git:** Download and install from git-scm.com.
- (Optional) NVIDIA Driver + CUDA:
  - Needed only if you want to benchmark the model on a GPU. CPU inference works out of the box.

#### Model at a Glance

#### **Architecture**

Block	Details
Frame Encoder	1-D CNN (99 → 128 → 512 channels)
Temporal Encoder	2-layer Bi-LSTM (256 hidden units/direction)
Exercise Embedding	6 → 64 MLP

Block	Details
Prediction Heads	Quality (2-way) • 14-Joint Regression • Exercise ID (6-way)
Total Footprint	3.41 M parameters (≈ 13 MB)

#### **Performance Highlights**

- of 91.5% F1-score on repetition quality assessment.
- D 99.5% F1-score on exercise identification.
- 4.73° Mean Absolute Error for joint-angle regression.
- Ø ~30 FPS on a mid-range CPU (or ~7,500 windows/second on GPU).



This project is released under the **MIT License**. The dataset media (REHAB24-6) remain under their original license. Please see the <u>REHAB24-6 Zenodo record</u> for details.

## Citation

If you use RehabApp or its components in your research, please cite it as:

@misc{rehabapp2025,

```
author = {Jithin Krishnan},
title = {RehabApp: AI-Powered Real-Time Physiotherapy Coach},
year = {2025},
```

howpublished = {\url{https://github.com/IS06-Jithin/RehabApp}}