# FITLENS – INSTALLATION GUIDE AND USER MANUAL

**Version:** 1.0
**Project Type:** Body Measurement System Using Front + Side Images
**Tech Stack:** Python, Flask, YOLO, MediaPipe, OpenCV, JavaScript Frontend

## 1. Introduction

FitLens is a computer-vision–based body measurement system that extracts key anthropometric measurements from user-uploaded images. The system uses front and side body photos along with the user's height to compute accurate body proportions. The backend uses YOLO for segmentation, MediaPipe Pose for landmark detection, and OpenCV for processing. The frontend provides a simple, browser-based interface for uploading images and displaying results.

This document explains the full installation steps, environment setup, application structure, and usage instructions.

## 2. System Requirements

### 2.1 Hardware Requirements

- Minimum 4 GB RAM
- Recommended 8 GB RAM
- At least 2 GB of free disk space
- Webcam or mobile camera (for testing image capture)

### 2.2 Software Requirements

- Windows 10 or higher
- Python 3.9 – 3.12
- Node.js (if using frontend with npm)
- Git (optional)
- Visual Studio Code (recommended)

## 3. Project Structure

After extracting the provided project folder, the structure will appear as follows:

```
FitLens/
│
├── backend/
│   ├── app.py
│   ├── requirements.txt
│   ├── yolov8n.pt      (or other model file)
│   ├── utils/
│   └── processing/
│
├── frontend/
│   ├── index.html
│   ├── script.js
│   ├── styles.css
│
└── README.md
```

## 4. Backend Installation (Python + Flask)

### Step 1: Open the Project in VS Code

1. Extract the zip file.
2. Open Visual Studio Code.
3. Select "Open Folder" and choose the project directory.

### Step 2: Open a Terminal

In VS Code, open the terminal using:

Ctrl + `

### Step 3: Navigate to the Backend Folder

```
cd backend
```

### Step 4: Create a Virtual Environment

```
python -m venv .venv
```

### Step 5: Activate the Environment

```
.venv\Scripts\activate
```

### Step 6: Install Dependencies

```
pip install -r requirements.txt
```

Packages installed include:

- Flask
- OpenCV
- MediaPipe
- Ultralytics (YOLO)
- NumPy
- Pillow
- Werkzeug

**Step 7: Run the Backend Server**

```
python app.py
```

You should see:

```
Running on http://127.0.0.1:5000
```

This URL is your local backend API.

## 5. Frontend Installation

If your frontend is static (HTML/CSS/JS), no installation is required.

**Step 1: Navigate to the Frontend Folder**

```
cd ../frontend
```

**Step 2: Open index.html**

Right-click → "Open with Live Server" (if you installed Live Server extension).

If Live Server is not used, open the file directly in the browser:

file:///your-path/frontend/index.html

The frontend will communicate with the backend using:

http://127.0.0.1:5000

## 6. How the System Works

### 6.1 Image Inputs

The user provides:

- One front-facing full-body image
- One side full-body image
- Height in centimeters

### 6.2 Processing Pipeline

1. YOLO model performs body segmentation.
2. MediaPipe Pose extracts body landmarks (shoulders, hips, knees, etc.).
3. Pixel measurements are converted to real-world values using height as the scale reference.
4. Calculated measurements are returned as JSON to the frontend.

### 6.3 Output Measurements

The system computes estimated values for:

- Shoulder width
- Chest circumference
- Waist circumference
- Hip circumference
- Arm length
- Leg length
- Total body height (as verification)

## 7. User Guide

### Step 1: Open the Frontend

Launch index.html through Live Server or by opening it in the browser.

### Step 2: Enter Your Details

Enter:

- Height (cm)
- Front image
- Side image

### Step 3: Submit

Click the "Get Measurements" button.

**Step 4: Wait for Processing**

The system sends data to the backend and processes the measurements.

**Step 5: View Results**

Measurements will be displayed on-screen.
Values can be copied or exported depending on interface options.

### 8. Troubleshooting

**Issue: pip install fails**

Cause: Internet timeout or corrupted cache.
Solution:

```
pip install --default-timeout=200 -r requirements.txt
```

**Issue: YOLO model not found**

Ensure yolov8n.pt or your chosen model file is placed inside the backend folder.

**Issue: CORS error**

Enable CORS inside app.py (if required):

```
from flask_cors import CORS
CORS(app)
```

**Issue: 500 Internal Server Error**

Check that image files are correctly uploaded and paths exist.

## 9. Notes for Deployment (Optional)

For deployment:

- Backend can be deployed on services like Render or AWS EC2.
- Frontend can be hosted on GitHub Pages, Netlify, or Vercel.
- Use HTTPS for secure communication.
- Ensure proper CORS configuration.

## 10. Conclusion

FitLens provides a robust computer-vision pipeline for estimating anthropometric body measurements from images. Using YOLO for segmentation and MediaPipe for pose detection allows the system to measure users accurately using only images and height information. This guide details the installation, usage, and troubleshooting steps required for smooth execution of the application.