Problem to solve: Given a camera feed, a user can be in front of the camera and the application should be able to determine the clothing size of the person (body measurements).

Breaking down the problem into tasks:

1. Generate a camera feed for the user.
2. Identify the relevant body parts of the user required for clothing size measurements.
3. Calibrating the camera with the movement of the user (distance between the user and camera).
4. Determining the actual measurements from virtual measurements (from the images produced by the camera).
5. Packaging the solution and making it deployment ready.

Task 1

- Create an interface that has a widget/element to encompass the camera feed.
- Take input such as height of the user for better and easier calibration.
- Calculate/determine the details of the camera such as field of view, focal length, etc.

Task 2

- The user can be made to stand at a certain distance from the camera, holding an object such as a checker-board to get a certain posture for better results.
- A rule-based approach can be used wherein a flat file (XML/JSON) can contain rules for determining body parts using frameworks like opencv.

Machine Learning approach

- The sub-problem in task 2 can be identified as a problem to detect body parts of the user.
- In Computer Vision, this problem can be solved using an approach known as 'Object Segmentation'.
- A deep learning model (CNN) can be trained on several images (human body data-set) on a task where the model takes a raw image as input and learns to generate a segmented image by labeling pixels of the same body parts (like hands, shoulders, waist, etc.) under common color code.
- Once the pixels are color coded, this becomes a classification problem to identify the labeled pixels as the respective body parts.

Using a pre-trained model

- A pre-trained model like BodyPix can be used to eliminate the overhead of training a model from scratch.
- BodyPix is a pre-trained model that determines the different body parts (with a part id from 0-24).
- The image of the person on the feed can be segmented easily and a linear distance between the body parts like left shoulder & right shoulder can be determined using the tensorflowjs APIs.

Task 3 & 4

- The user's height, as entered, is used to calibrate distances in the camera field-of-view.
- Once parts of the user's body are identified, the calibrated (virtual) scale is used to measure distances between key points of the body part to determine dimensions of relevance to clothing size.
- The ratio of the focal length and the distance between the user and the camera and the ratio of the length of the virtual scale (line segment) and the actual body part length can be used to identify the real length of the body part for clothing size measurement.
- The data can be captured in real-time on the User Interface.
- This can also be built into a micro-service which can be invoked and it will return the request object with the measurements of the relevant body part (as requested by the tailor).

Task 5

- The UI can be treated as the client and the model processing can be treated as the server (endpoint to be defined).
- The dependencies file and tests can be created to validate the e2e working of the entire application.
- The latency and accuracy of the model can be captured to enhance the performance by optimization.
- Build and run scripts can be created for a smooth run and deployment.

References:

- [BodyPix – Person Segmentation in the Browser](#)
- [[Updated] BodyPix: Real-time Person Segmentation in the Browser with TensorFlow.js](#)
- [Real-time Clothing size body measurement estimator using TensorFlow.js](#)
- [Real-time Clothing Size Estimation Using Body Segmentation](#)