Work It

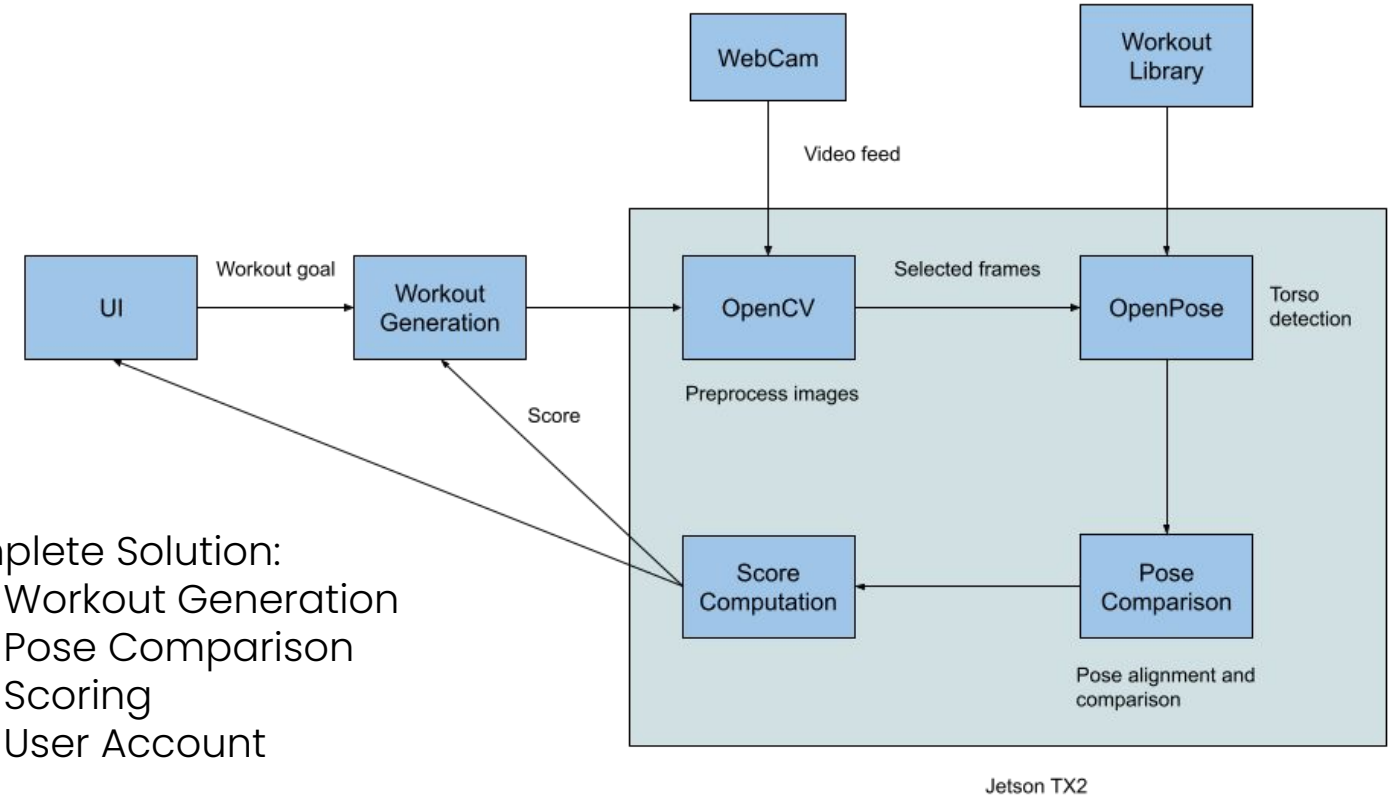
MADELINE MIANZO
SARAH TAN
ZIXUAN ZOU

1. Generate user specific workouts to cater to their fitness level

2. Use Tensorflow OpenPose to analyze body position of the model and the user

3. Give the user a score for their workout

Block Diagram



Complete Solution:

- Workout Generation
- Pose Comparison
- Scoring
- User Account

Software & Hardware

Generate workout sequence based on user's workout goal and the fitness level

Workout Generation

OpenCV
- Preprocessing
- Frame Selection

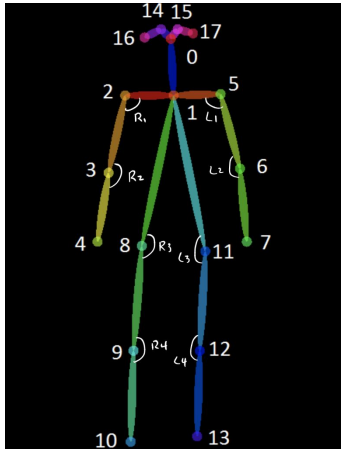
OpenPose
- Torso Detection
- Calibration

Pose Comparison
- Pose Alignment

Score Computation

Compute the user's score based on the comparison results

Evaluate the user's fitness level based on the score and adjust the difficulty level



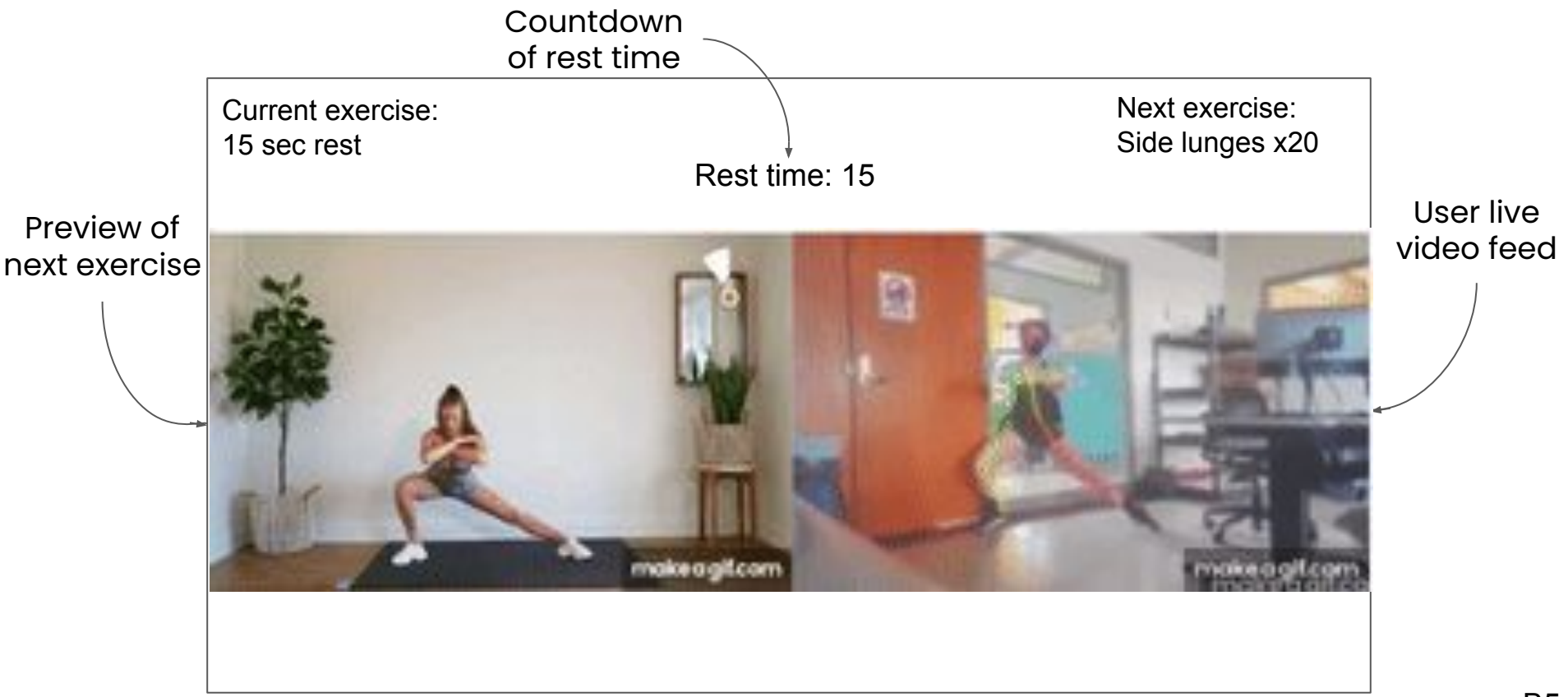
WebCam

Laptop
- UI
- Workout Generation
- Workout Library

NVIDIA Jetson TX2
- OpenCV
- OpenPose
- Posture Analysis

System Specification - UI

System Specification - UI



Pose Comparison

	A	B	C	D	E	F	G	H
1	workout	frame_numb	timestamp	duration	total_frame	coordinates	angles	
2	arm1_1	3	[4, 11, 23]	1.7	27	[[(231, 60), (	[[2.4227626539681686,	
3	arm1_2	2	[1, 9]	0.8	14	[[(231, 60), (	[[2.9708574421145104,	
4	arm1_5	2	[1, 18]	3	44	[[(316, 225),	[[1.6718996985311951,	
5	arm1_6	4	[1, 30, 53, 75	6.2	94	[[(158, 182),	[[0, 0, 2.5884951166780	
6	leg1_1	2	[1, 35]	5	77	[[(267, 134),	[[1.317902261699299, 1	
7	leg1_2	2	[1, 23]	2.5	41	[[(200, 56), (	[[0.7378150601204649,	
8	leg1_3	2	[5, 29]	2.1	41	[[(227, 66), (	[[0.4831256648173338,	
9	leg1_2.1	2	[1, 17]	2.2	36	[[(273, 57), (	[[2.8883268873590406,	
10	leg1_4	3	[8, 20, 43]	3	46	[[(256, 62), (	[[3.004526479279605, 0	
11	core1_1	3	[1, 23, 62]	6	79	[[(158, 207),	[[2.5207447236329314,	
12	core1_4	1	[19]	2.2	34	[[(193, 200),	[[0, 0, 1.7089392929813	
13	arm1_8	3	[1, 31, 91]	7.4	112	[[(280, 157),	[[0, 0, 0, 0, 1.986510201	
14	leg1_5	2	[1, 30]	4.1	52	[[(160, 104),	[[0.47835243137865757	
15	leg2_2	2	[1, 18]	2.4	37	[[(238, 59), (	[[0.5328438876193887,	
16	leg2_3	3	[1, 18, 51]	4	63	[[(247, 60), (	[[1.7625844687816017,	
17	arm1_5.1	2	[1, 23]	2.8	44	[[(196, 178),	[[1.406374921979318, 2	
18	arm1_13	2	[1, 45]	5.4	81	[[(187, 69), (	[[3.141592653589793, 2	
19	core2_5	4	[1, 29, 55, 83	7.2	108	[[(160, 191),	[[0, 0, 2.3482790882236	
20	core1_11	3	[1, 18, 45]	3	57	[[(164, 219),	[[0.3332443011116726,	
21	arm2_5	3	[1, 23, 62]	4.7	72	[[(342, 176),	[[2.203214145589375, 3	
22	arm2_6	2	[6, 12]	0.7	12	[[(329, 186),	[[0.8760580505981935	

Metrics

Requirements	Testing	Metrics
Hardware Performance	Time how long it takes to analyze sets of images	< 1 min time limit
OpenPose Detection	Analyze runtime and accuracy for different image sizes/poses	90% accuracy
Pose Alignment	Analyze comparison results over different body types	90% accuracy
Pose Comparison	Test with similar poses/workout exercises	90% accuracy
Score Computation	Analyze the scores over different levels of completion	Score should reflect user's completion and accuracy

OpenPose Testing

Model	Avg Runtime(s)	Accuracy
CMU	1.14478	86%
Mobilenet_thin	0.23142	42%
Mobilenet_v2_large	0.23196	60%
Mobilenet_v2_small	0.16870	36%

Number of images = 50, Resize = 432x368

Workout	Avg Runtime(s)	Accuracy
Elbow to Knee	1.08485	70%
Rotating T Plank	1.08589	100%
Russian Twist	1.09047	90%
Side Lunges	1.09385	100%
Standing Extension	1.09028	90%

Model = CMU, Number of images = 10 per workout, Resize = 432x368

CMU Mobilenet_thin

Pose Comparison Testing

Frame Size	Avg Accuracy
160x96	72%
432x368	90%
656x432	88%

Workout	Avg Threshold (radian)	Avg Accuracy
Arm	7.4	91.67%
Core	11.87	76.19%
Leg	9.12	94.12%

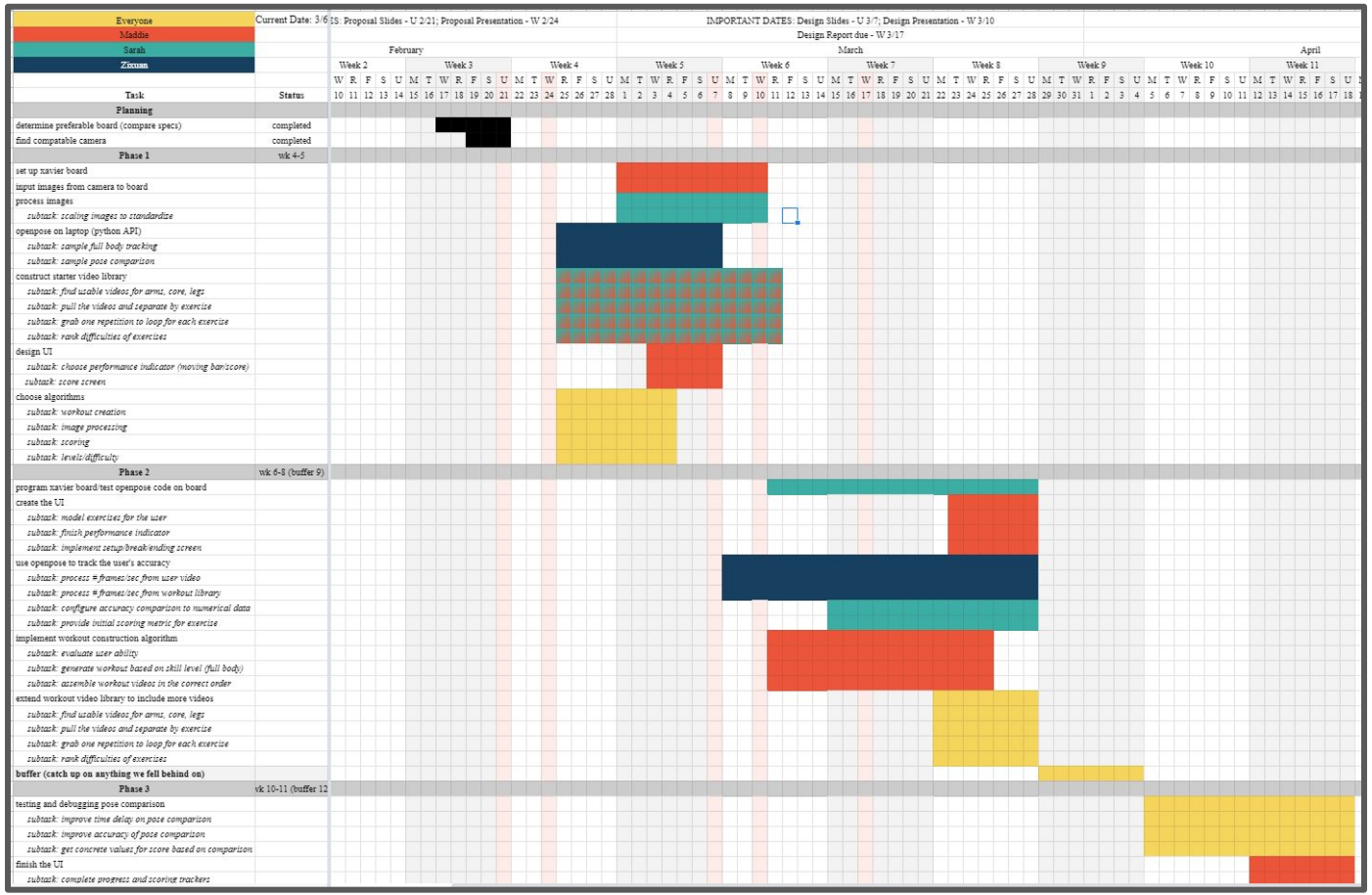
Pose Comparison Testing

	Full Repetitions (expected)	**Full Repetitions (actual)**	**Partial Repetitions** (50% of **expected** amount)	**Partial Repetitions (actual)**
Good Form ('User video' is a similar YouTube clip of same exercise)	Base score	600	≈ Base score * 0.5	300
Poor Form (Ex: side lunges that only go down to 45° bend that should be 90°)	≈ Base score * 0.5	400	<= Base score * 0.25	100

Challenges

- Dependencies on Xavier board
  - Tensorflow
- Ensuring UI and pose comparison code work in parallel
- Initial testing didn’t have a suitable background
  - Lighting
  - Plain background
- OpenPose detection for some exercises

Gantt Chart


Everyone
Maddie
Sarah
Zixuan
Current Date: 3/6
IMPORTANT DATES: Proposal Slides - U 2/21; Proposal Presentation - W 2/24
IMPORTANT DATES: Design Slides - U 3/7; Design Presentation - W 3/10
Design Report due - W 3/17
February
March
April
Week 2
Week 3
Week 4
Week 5
Week 6
Week 7
Week 8
Week 9
Week 10
Week 11
Task
Status
Planning
determine preferable board (compare specs)
completed
find compatable camera
completed
Phase 1
wk 4-5
set up xavier board
input images from camera to board
process images
subtask: scaling images to standardize
openpose on laptop (python API)
subtask: sample full body tracking
subtask: sample pose comparison
construct starter video library
subtask: find usable videos for arms, core, legs
subtask: pull the videos and separate by exercise
subtask: grab one repetition to loop for each exercise
subtask: rank difficulties of exercises
design UI
subtask: choose performance indicator (moving bar/score)
subtask: score screen
choose algorithms
subtask: workout creation
subtask: image processing
subtask: scoring
subtask: levels/difficulty
Phase 2
wk 6-8 (buffer 9)
program xavier board/test openpose code on board
create the UI
subtask: model exercises for the user
subtask: finish performance indicator
subtask: implement setup/break/ending screen
use openpose to track the user's accuracy
subtask: process # frames/sec from user video
subtask: process # frames/sec from workout library
subtask: configure accuracy comparison to numerical data
subtask: provide initial scoring metric for exercise
implement workout construction algorithm
subtask: evaluate user ability
subtask: generate workout based on skill level (full body)
subtask: assemble workout videos in the correct order
extend workout video library to include more videos
subtask: find usable videos for arms, core, legs
subtask: pull the videos and separate by exercise
subtask: grab one repetition to loop for each exercise
subtask: rank difficulties of exercises
buffer (catch up on anything we fell behind on)
Phase 3
wk 10-11 (buffer 12
testing and debugging pose comparison
subtask: improve time delay on pose comparison
subtask: improve accuracy of pose comparison
subtask: get concrete values for score based on comparison
finish the UI
subtask: complete progress and scoring trackers