AI Fitness Trainer

***Author:*** *Aminu Abdusalam* ***Status:*** *In Works* ***Last Updated****: 8/19/2022*

# Overview

## Background

This project serves as an opportunity to enhance learning and exposure to the exciting computer vision field. Today, we see our technologies like computer vision can be applied in automating our everyday activities and the process of getting fit is not excluded. For instance, taking on activities like bouldering and hiking to stay fit was not a huge success for me. If I had an AI Fitness Trainer, accomplishing my fitness goals would have been much easier. Thus, this project is a win-win, as the developer gets to enhance their experience while working on something they are super interested in, and the users get a fantastic product that satisfies their need.

## Goals

**TL; DR:** The goal of the project can be summarized as developing an **AI fitness trainer** embedded with **storage and recommender systems** and an **AI virtual mouse**.

The AI fitness trainer will help the user lose weight, gain muscle, and accomplish other fitness goals. In addition, it'd attempt to understand the client goals, develop a fitness routine, recommend a healthy eating plan, and ensure all exercises are performed correctly.

## Skills & Technologies

The following skills and technologies are required and will be learned for the project:

**Skills:** Computer Vision, Pose Landmarks, Pose Estimation, Hand Tracking

**Technologies**: Python, OpenCV, MediaPose, Streamlit, Nylas (ML-Powered Communications Tool)

## Team

|  |  |
| --- | --- |
| **Role** | **Name** |
| **PM & Dev** | *Aminu Abdusalam* |
| **Mentor** | *Dr. Lei Qian* |

# Requirements

This section lists all the requirements to be completed for the project and their priority levels (**0** – must have, **1** – partial progress must have, **2** – nice to have if time permits).

|  |  |  |
| --- | --- | --- |
| **Requirement** | **Description** | **Priority** |
| Develop AI Trainer | Implement Pose Detection i.e. detect body movements in a video stream using ML algorithms | 0 |
| Build Trainer using Pose Detector to ensure exercises are performed correctly. | 0 |
| Build UI | Design website via Figma | 0 |
| Build web app using Streamlit framework. | 0 |
| Build Storage and Recommender System | Enable user login and Authentication. | 0 |
| Send report of activities and related information to user after every session (total calories burned, total workout time, etc.) | 0 |
| Store and keep track of unique user data (e.g. target calories, total number of sessions (fitness training days), total calories burned (and per month), number of exercises or days to target calories, etc.) | 1 |
| Recommend fitness routine and healthy eating plan based on user’s goals | 2 |
| Build AI Virtual Mouse | Implement Hand Tracking | 2 |
| Build Virtual Mouse using Hand Tracker to allow user control machine without touching the machine | 2 |

# Timeline

This section lists the milestones of the project spread across two semesters (Fall 2022 and Spring 2023).

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestones** | | **ETA** | **Status** |
| **Requirements Gathering** (Project Idea, Project Proposal) | | 08/22 | Completed |
| **Design Exploration** (Setup and Installation of necessary technologies, Addition of Project to remote repo, Skill Preparation, Framework Project) | | 08/29 | Completed |
| **Implementation** | Complete Pose Estimation Build | 09/19 | Completed |
| Complete AI Personal Trainer for **Curls** | 10/11 | Completed |
| Complete AI Personal Trainer for **Pushups** | 10/18 | Completed |
| Complete AI Personal Trainer for **Squats** | 10/25 | Completed |
| Complete speech-to-text et text-to-speech | 10/31 | Completed |
| **Quality Testing** | | 11/7 | Completed |
| **Midpoint Presentation Draft** | | 11/14 | Completed |
| **Midpoint Demo & Report** | | 11/21 | Completed |
|  | **WINTER BREAK** | | |
| **Project Review** (Current status and Re-evaluation of Next Steps as Needed) | | 01/16 | Completed |
| **Implementation** | Complete FigJam design for the Web app | 01/22 |  |
| Build Web app with only one exercise type option | 01/22 |  |
| Augment website with multi pages to allow for selection of exercise choice | 02/05 |  |
| Implement “about developer and project” page including instructions on how to use product | 02/12 |  |
| Develop a “report” page that showcases details about their fitness sessions at the end of each session including an option to send these details to their mail | 02/20 |  |
| Add authentication/login capabilities | 03/05 |  |
| Complete Hand Tracking Build (Stretch Goals) | 03/13 |  |
| Complete AI Virtual Mouse Build (Stretch Goals) | 03/27 |  |
| **Quality Testing (**arrange project files and ref., test project) | | 04/3 |  |
| **Final Presentation Draft** | | 04/10 |  |
| Final Demo & Report | | 04/17 |  |

# References

* Presentation:
  + Part 1: <https://1drv.ms/p/s!AlGAL8aQVodopGfx1njotqcRKHrS?e=YRk7SU>
* WebApp FigJam Design: [AI-FitnessTrainer - FigJam (figma.com)](https://www.figma.com/file/aqxyqorcf4DUSFmENwS3Hl/AI-FitnessTrainer?node-id=0%3A1&t=bt59qYdskhjIeFAl-0)
* Online Course:
  + [Advanced Computer Vision with Python - Full Course](https://www.youtube.com/watch?v=01sAkU_NvOY&t=16511s)
  + [Advance Computer Vision with Python - Computer Vision Zone](https://www.computervision.zone/courses/advance-computer-vision-with-python/)
* Similar Products: [15 Computer Visions Projects You Can Do Right Now - neptune.ai](https://neptune.ai/blog/15-computer-visions-projects)
* Study Material: [Pose Classification Options  |  ML Kit  |  Google Developers](https://developers.google.com/ml-kit/vision/pose-detection/classifying-poses)
* [Markdown reference for docs.microsoft.com - Contributor Guide | Microsoft Docs](https://docs.microsoft.com/en-us/contribute/markdown-reference)
* Trainer Videos: Pexels.com

# Open Questions

|  |  |  |
| --- | --- | --- |
| **Description** | **Next Steps** | **Status** |
| What are landmarks? | In many computer vision applications, neural network often needs to recognize essential points of interest (than bounding box) in the input image. We refer to these points as landmarks. In such applications, we want the neural network to output coordinates (x, y) of landmark points than those of bounding boxes. |  |
|  |  |  |