AI Fitness Trainer

*Progress Report #18*

# TL; DR

**On track (*Github:***[*AI-FitnessTrainer*](https://github.com/aminuabdusalam/AI-FitnessTrainer)*)***.** Continued build of webapp by **03/18.**

# Project Goals (Recap)

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**.

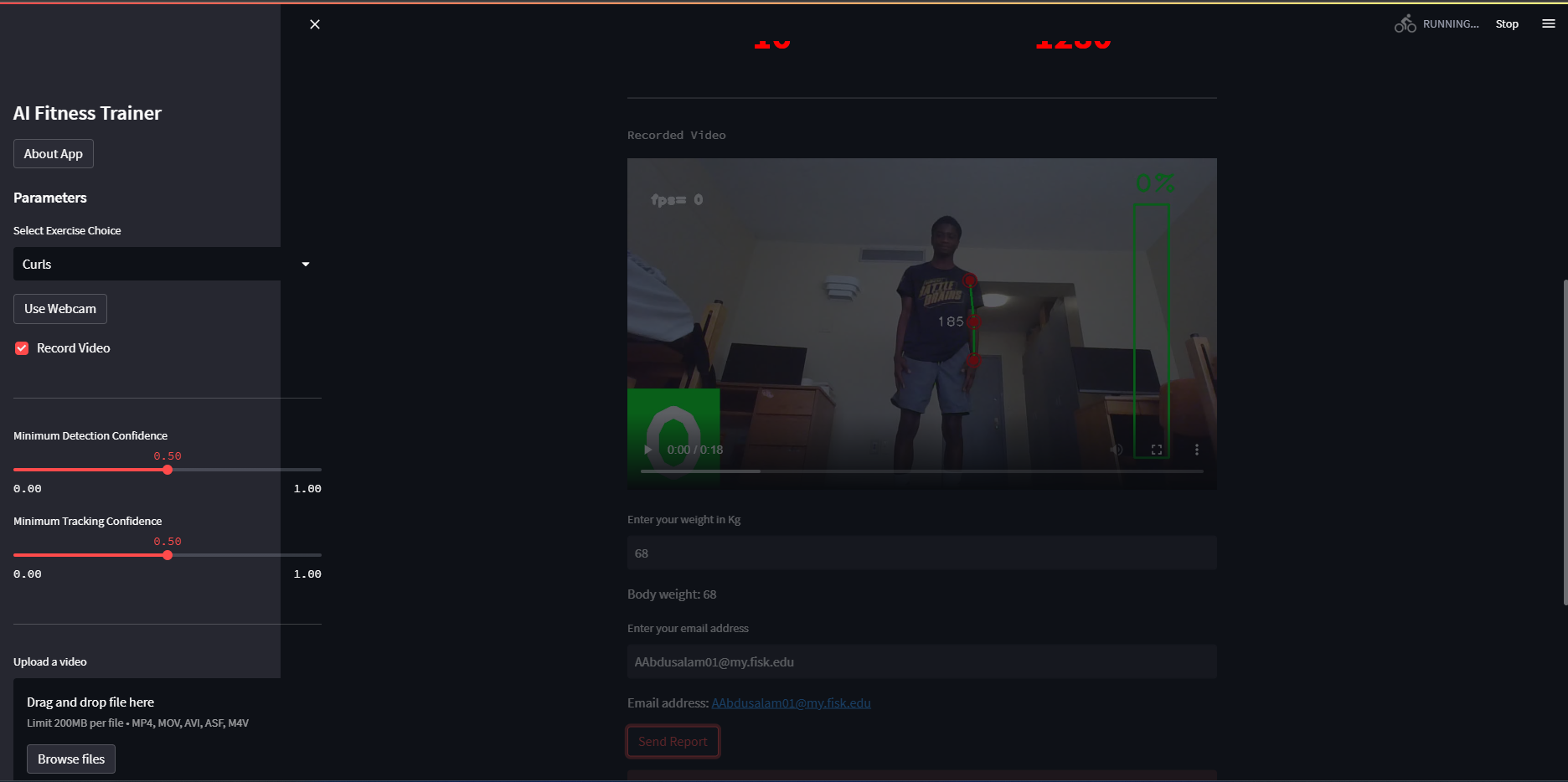
The AI fitness trainer will help the user lose weight, gain muscle, and accomplish other fitness goals. In addition, it'd attempt to understand the client goals, develop a fitness routine, recommend a healthy eating plan, and ensure all exercises are performed correctly.

# Highlights

* Continued build of Webapp.

[feature: implemented report sending by aminuabdusalam · Pull Request #11 · aminuabdusalam/AI-FitnessTrainer (github.com)](https://github.com/aminuabdusalam/AI-FitnessTrainer/pull/11)

* + Utilized python **smtplib and email** libraries to implement “email sending” feature. This included generating an app-specific password for my email account in order to authenticate with the SMTP server.
  + Separated the input box rendering from the rest of the app's logic to ensure that the input box is only rendered once, and that changes to its value do not trigger unnecessary reruns of main. The image below shows the page with the input text boxes and send report button. Note that we are using **Streamlit's** session state system to store the value of the input box. This allows us to persist the value between reruns of the app, even if the user refreshes the page or closes and reopens the app.

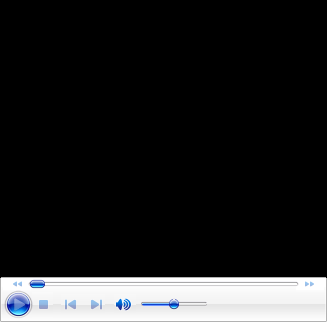


* + The “email-sending feature” in combination with the input text boxes for user’s body weight and email address allows a report to be sent to the use after every session. The burned calories is calculated with the formula: **[(MET value x 3.5 x body weight in kg) / 200] x time in minutes,** where MET value is a measurement of the intensity of the exercise, and body weight is in kilograms.

**Graphical user interface, text, application, email

Description automatically generated**

**Click Windows Media to Watch Demo:**



# Lowlights

None

# Next Steps

* Continue Web app build.
  + Fix issues with submission of text\_input causing reloading of app\_mode by **03/20**
  + Add authentication/login capabilities by **03/27**
  + Clean up frontend file i.e. add more comments and functions by **03/27.**

# Timeline

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Trainer service build completed** | | 11/21 | Completed |
| **WINTER BREAK 2022** | | | |
| **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 | Completed |
| Build Web app with only one exercise type option | 01/22 | Completed |
| Augment website with multi pages to allow for selection of exercise choice | 02/05 | Completed |
| Implement “about developer and project” page including instructions on how to use product | 02/20 | Completed |
| 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 | 03/20 | **In Progress** |
| Add authentication/login capabilities | 03/27 |  |
| **Quality Testing** (arrange project file and refs, test project) | | 04/3 |  |
| **Final Presentation Draft** | | 04/10 |  |
| Final Demo & Report | | 04/17 |  |