



UNIVERSITY OF  
**LEICESTER**

## **Boxing Training & Improvement App**

CO3201 Computer Science Project Interim Report

School of Computing and Mathematical Sciences,  
University of Leicester

**By Hugo Hull,**

November 2023

# Table of Contents

Declaration.....	3
1. Aims & Objectives.....	4
1.1. Aims.....	4
1.2. Objectives.....	4
2. Background Research.....	5
2.1. Existing Solutions.....	5
2.1.1. PunchLab Bag-tracking strap.....	5
2.1.2. FightCamp Wrist Trackers.....	6
3. Requirements.....	7
3.1. Essential Requirements.....	7
3.2. Optional Requirements.....	7
4. Outline of specification & Design.....	7
4.1. High-level Overview.....	7
4.2. Libraries.....	8
4.2.1. Tkinter.....	8
4.2.2. OpenCV.....	9
5. Planning & Timescales.....	10
6. Bibliography.....	12

## Declaration

All sentences or passages quoted in this report, or computer code of any form whatsoever used and/or submitted at any stages, which are taken from other people's work have been specifically acknowledged by clear citation of the source, specifying author, work, date and page(s).

Any part of my own written work, or software coding, which is substantially based upon other people's work, is duly accompanied by clear citation of the source, specifying author, work, date and page(s).

I understand that failure to do this amounts to plagiarism and will be considered grounds for failure in this module and the degree examination as a whole.

**Name:** Hugo Hull

**Signed:**

A handwritten signature in black ink, appearing to read 'Hugo Hull', written in a cursive style.

**Date:** 24/11/23

# 1. Aims & Objectives

## 1.1 Aims

The aim of this project is to develop and create a desktop application that allows users to improve and train their boxing skills from home. The application will generally focus on a live punch video tracking feature, a feature which has not yet been implemented in a boxing application. Implementing the feature in this way will allow users to improve their boxing skill in a way that previously would mean using a specific piece of external and costly hardware.

With the WHO estimating that 1 in 8 people suffer from a mental health condition [1], mental health issues are an ever present and rising problem within the worldwide population. Boxing and exercise in general are a known and well tested method to treat people with such conditions. In a study conducted by J. Bozdarov [2], boxing has been shown to provide “significant reduction in symptoms of anxiety, depression, PTSD and negative symptoms of schizophrenia”. This project aims to provide users with training that has traditionally only been achievable with a dedicated boxing coach, but at the comfort of their home. In turn helping to improve their mental health condition and their boxing skill simultaneously.

## 1.2 Objectives

1. A requirements elicitation will occur to understand what features will be implemented in the project.
2. Background research will be performed in order to understand what existing software does well and what could be improved for my application.
3. A list of essential and optional requirements will be produced stemming from the background research.
4. A rough sketch of the application will be to guide the creation of the GUI.
5. Expand knowledge of Python and its libraries in order to implement the features identified from the requirements elicitation and background research.
6. A basic GUI for the desktop application will be created using Python [3] and the library Tkinter [4].
7. A backend will be developed using Python that implements the basic functionality of the application.
8. The GUI for the live video punch tracking will be developed.
9. Using the Python library OpenCV [5], a live video punch tracking feature will be implemented.
10. Optional requirements found during the requirements elicitation will be developed on the front and back end.
11. Thorough testing will occur to ensure there are no major bugs within the software.

## 2. Background Research

Certain aspects of what I am aiming to achieve with this application will be a first on the market. However, where aspects have been implemented in other applications, I will aim to improve upon the idea and implement it in the simplest possible way for the user.

### 2.1. Existing Solutions

There are two existing ways of tracking punches with apps: bag-tracking straps and wrist trackers. I will discuss the implementation of both methods using the market leaders from each category, PunchLab and FightCamp respectively, both of which have dedicated mobile apps. There are currently no desktop applications that offer punch tracking or tracking using a camera.

#### 2.1.1 PunchLab Bag-tracking Strap



PunchLab [6] is a mobile application that aims to improve the user's boxing ability with "Boxing workouts you can do at home" [6]. The app aims to provide people with workouts they can follow along to at any level. This is good as it allows users of all skill levels to be involved and to get healthier. The app also features an activity page where the user can see statistics over how they've performed in the past. Many users have found this to be an effective way to track, and in most cases, improve consistency. This is something I may consider including into my application in the future.

PunchLab also offers the bag-tracking strap as an optional, costly extra [7]. The strap costs £36.99 but does not include the yearly membership to their premium app which costs £70 a year. This is very expensive for a beginner to get involved with and requires them to have access to a boxing bag to get use out of the item. Not to mention this does not include boxing gloves or wrist wraps both of which are necessary items. All these costs make the barrier of entry for a beginner very high and may put them off training and using an item like this all together. However, I can see that for someone with more experience this may be a great option for them.

My application addresses this by severely lowering the barrier of entry. If my app were to go into production, all that would be necessary is a laptop or pc and a pair of boxing gloves. There is no need for a boxing bag with this solution. While the cost would have to be deliberated, it would be considerably less than the PunchLab setup and requires less equipment.

Another downside is that you must strap your mobile device to the bag itself while you are using this item. Some people may not be comfortable with this and would prefer that their device is far away. For these people my application would be a better idea because the device you're using to record the live footage does not need to be close to the punch bag or area where the shadow boxing is taking place.

### 2.1.2 FightCamp Wrist Trackers



FightCamp [8] is a mobile application that aims to improve the user's boxing ability. The app works by using wrist trackers [9] to track the punches thrown by the user. This solution is better than the PunchLab bag-strap as a bag is not necessary for the tracking feature to work. This lowers the barrier of entry massively. This makes the FightCamp have a similar barrier of entry to my app's solution, however, the wrist trackers cost \$99 as well as a paid subscription-based app.

Again, like with the PunchLab strap, the cost is very high and may be off putting to a beginner who would not want to spend a massive amount of money for something that they may not get a benefit out of. My app has a huge advantage over these two solutions in this area.

Compared to the PunchLab app, the FightCamp workouts are not quite as informative or good quality. This is something that has been reflected in the reviews of users and well-known fitness influencers. When developing my app, I now know and value the importance of having easy to follow and good quality exercise tutorials. This is something I will be sure to implement.

## 3. Requirements

The requirements have been split into two categories: essential (requirements that must be complete for the application to function) and optional (requirements that are not essential to the function but would greatly improve the quality of the app). As this is a live project, the requirements are subject to change over the course of development.

### 3.1 Essential requirements

1. A user can start a boxing round timer with full control over the length of the round and the rest, as well as number of rounds.
2. A user can select an option for the round timer where during the round, punch combinations are audibly played.
3. A user can use a webcam or camera for live punch tracking in the app.
4. A user can see their history including the number of punches thrown and the number of rounds completed.

### 3.2 Optional requirements

1. A user can see a more detailed history which may include number of left and right punches thrown and the number of specific combinations thrown.
2. A user can choose to record live video and store it within the application.
3. A user can see the speed and distance travelled of specific punches during the live video tracking.
4. A user can see tutorials for basic techniques shown via a video/animation.
5. A user can manually input the number of rounds and minutes of sparring completed.
6. A user can see a list of example exercises e.g. skipping, pushups etc.
7. A user can make notes in their history so that they can track specific sparring sessions and details around the session for example.
8. An AI response can analyse the history of the user and create detailed charts including punches thrown, number of rounds and other metrics that are tracked.

## 4. Outline of Specification & Design

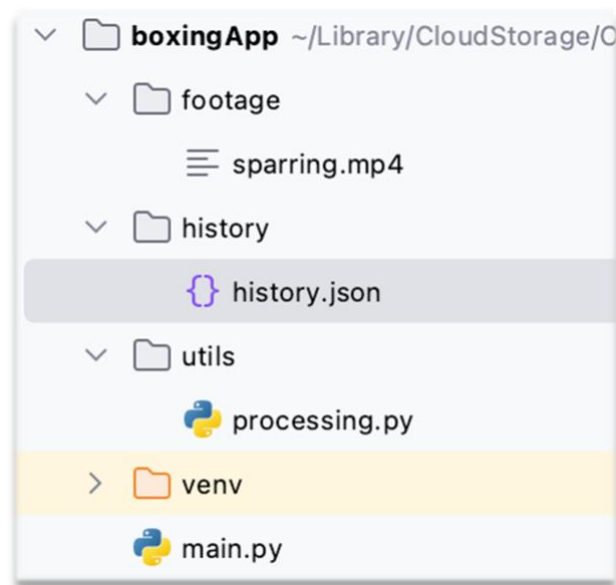
### 4.1 High-level Overview

The project will be developed entirely in Python using various libraries. The main.py will have the code responsible for the GUI inside of it and is responsible for handling the user's inputs. All the python files responsible for the image processing necessary for the live punch tracking will be stored inside of a 'utils' folder.

Video recorded by the user will be stored inside of a file named 'footage'. The app will

then access the footage from there, making it a quick response time. The history of a user will be stored inside of a JSON file, this is done because a JSON is lightweight and all that as to be done to retrieve the data is to read the JSON. All relevant files to the user will be stored inside of the application. I have decided to this because the application is already going to be a desktop application so there's no need for a service such as a cloud server.

Here is a high-level overview of the structure I just described:



## 4.2 Libraries

In this section, I will describe the Python libraries that I will use to develop my project.

### 4.2.1 Tkinter

Tkinter is an integral part of Python's standard library, known for its role in creating graphical user interfaces (GUIs). Tkinter provides Python developers with a simple and effective way to create GUI applications. Tk, has been seamlessly integrated with Python, offering a diverse range of widgets and tools for building customizable interfaces.

The inception of Tkinter dates back to Python's early days, making it one of the most long-standing GUI frameworks in the Python ecosystem. Despite its age, Tkinter has remained relevant and widely used due to its simplicity and the fact that it comes pre-installed with Python, eliminating the need for additional downloads or installations.

Key Features of Tkinter:

- **Cross-Platform Compatibility:** Just like Python, Tkinter applications are inherently cross-platform, running seamlessly on all systems. This feature is particularly



valuable for my app as it can be deployed across all platforms without the need for recoding.

- **Extensive Widget Library:** Tkinter comes with a comprehensive set of widgets, including buttons, labels, text boxes, menus, and canvases. These elements can be easily customized and manipulated to build functional and visually appealing interfaces.
- **Customizability and Flexibility:** While offering simplicity, Tkinter also allows for considerable customization, enabling developers to create unique GUI layouts and designs.
- **Lightweight and Responsive:** Tkinter applications are typically lightweight, contributing to faster load times and responsiveness, an essential aspect of user experience.

However, like any technology, Tkinter has its drawbacks:

- **Limited Modern Features:** Compared to more modern GUI frameworks, Tkinter lacks certain advanced features and may not be the best choice for highly complex or visually intricate applications.
- **Aesthetic Limitations:** The visual aspects of Tkinter widgets are often considered basic.

As a whole, Tkinter is perfectly suited for a small sized project like my app. It will not need cutting edge design features as the simplistic GUI will make the app very easy to use and navigate for the user.

#### **4.2.2 OpenCV**

OpenCV (Open Source Computer Vision Library) is a highly acclaimed, open-source computer vision and machine learning software library. OpenCV's primary focus is on real-time image processing, which is critical in a wide range of applications, including interactive art, mines inspection, stitching maps on the web, or through advanced robotics.

OpenCV has grown to include over 2500 optimized algorithms. These algorithms encompass a broad spectrum of functions including facial recognition, object identification, classification of human actions in videos, tracking camera movements, tracking moving objects, extracting 3D models of objects, and much more.

Key Features of OpenCV:

- **Robust Library:** OpenCV boasts an extensive collection of algorithms for image processing and computer vision which is necessary for the punch tracking.
- **Cross-Platform Compatibility:** OpenCV supports various platforms, including Windows, Linux, macOS, and Android.

- **Real-Time Processing:** OpenCV is optimized for real-time applications, offering efficient ways to perform image processing and computer vision tasks. This feature is crucial for the live punch tracking feature of my app.
- **Community and Support:** OpenCV benefits from a large community of developers and users, providing extensive resources, documentation, and forums for troubleshooting and learning.

For my project, the real time analysis will be key as OpenCV can process and analyse images in real-time, which is essential for tracking the movement of a punch during training sessions. Also, the motion tracking and analysis algorithms OpenCV provides, can track the gloves speed and distance, meaning these metrics can be relayed to the user.

However, some drawbacks include:

- **Learning Curve:** OpenCV can be complex to learn and implement effectively, meaning I may have to take a lot of time to learn how to use the library effectively.
- **Performance Optimization:** To achieve real-time performance, especially for high-resolution videos, I may need to optimize my code and possibly use hardware acceleration.

Overall, for my application, OpenCV was the best choice due to the extensive support available as well as providing all the necessary functionality and more that my app will need. The support available I hope will help to offset the large learning curve that OpenCV has.

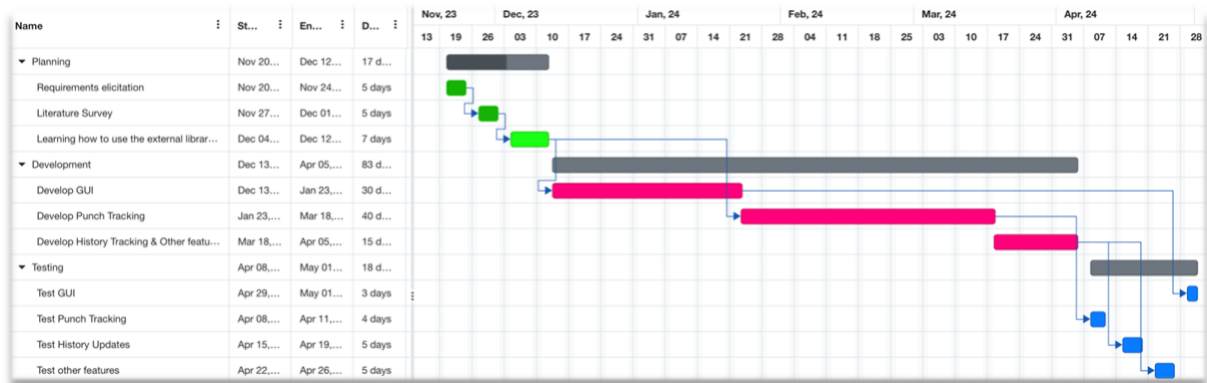
## 5. Planning & Timescales

The first semester is dedicated predominantly to the planning and design phases of the project. Achieving proficiency in Tkinter, Python, and OpenCV is crucial for the success of the project. Therefore, a substantial portion of this semester will be allocated to gaining a comprehensive understanding of these libraries and the Python language. This foundational period is essential for ensuring a solid base upon which the rest of the project can be effectively built upon.

In the second semester, the focus will shift to the development stage, which constitutes the majority of the project's timeline during this period. This phase involves the practical application of the skills and knowledge acquired in the first semester. It's a time-intensive process, requiring great attention to detail and the integration of various aspects of Tkinter, Python, and OpenCV. The development stage is pivotal in transforming the conceptual designs of the first semester into a tangible, functioning system.

After development is achieved, testing will take place to ensure that the project has no major bugs that affect the function of the app.

I will present a detailed Gantt chart to outline my projected timeline for the project, highlighting the anticipated milestones and their completion dates. This chart aims to offer a practical and realistic overview of the project's progression over time. It thoughtfully represents the allocation of time for each task, factoring in the necessary period for skill development in the relevant technologies. This visual tool is designed to accurately reflect the time commitment required for each segment of the project, ensuring a comprehensive and well-planned approach to its execution.



## 6. Bibliography

- [1] WHO, “Mental Disorders”, link: <https://www.who.int/news-room/fact-sheets/detail/mental-disorders#:~:text=In%202019%2C%201%20in%20every,of%20the%20COVID%2D19%20pandemic> , 8/6/22.
- [2] J. Bozdarov, “Boxing as an Intervention in Mental Health: A Scoping Review”, Sage Journals, link: <https://journals.sagepub.com/doi/full/10.1177/15598276221124095>, 9/9/2022.
- [3] Python, link: <https://www.python.org>, 2023
- [4] Tkinter, link: <https://docs.python.org/3/library/tkinter.html>, 2023
- [5] OpenCV, link: <https://pypi.org/project/opencv-python/>, 2023
- [6] PunchLab SRL, PunchLab, Link: <https://punchlab.net>, 2023
- [7] PunchLab SRL, PunchLab Bag-strap, Link: <https://punchlab.net/getTheStrap> , 2023
- [8] FightCamp HYKSO INC, FightCamp, Link: <https://joinfightcamp.com> , 2023
- [9] FightCamp HYKSO INC, FightCamp Wrist Trackers, Link: <https://joinfightcamp.com/shop/products/fightcamp-trackers> , 2023