RHEINISCHE FACHHOCHSCHULE KÖLN

University of Applied Sciences

Studiengang: Informatik (Bachelor of Science)



**Transfer-Projekt**

Entwicklung eines Powerlifting-Trainingsrechner:

Berechnung von Trainingsintensitäten und

Vergleich mit Open Powerlifting Daten

Vorgelegt von: Luca Kursawe

651232018

Dozent: Prof. Dr. Susanne Rosenthal

Wintersemester 2023/24

# Projektbeschreibung

1. Ziel ist die Entwicklung eines Tools für Powerlifter, welches ihnen ermöglicht, die empfundene Intensität eines Trainings beziehungsweise eines bestimmten Trainingssatzes anzugeben und basierend darauf weitere Empfehlungen zu erhalten. Außerdem sollen die Daten der Open Powerlifting Website integriert werden, damit der Nutzer seine Leistung mit anderen Sportlern vergleichen kann. Die Benutzeroberfläche sollte nutzerfreundlich und modern sein und dem Nutzer deutlich die nötigen Informationen darstellen.
2. Für die Umsetzung des Projekts "RPE & Open Powerlifting Calculator" waren bestimmte Voraussetzungen notwendig. Dazu gehörten fundierte Kenntnisse in der Programmiersprache Python, insbesondere im Umgang mit Datenverarbeitung und grafischen Benutzeroberflächen. Des Weiteren war ein tiefes Verständnis für die spezifischen Bedürfnisse und Anforderungen von Powerlifting-Athleten erforderlich, um ein praxisnahes und effektives Tool zu entwickeln.
3. Die technischen Vorgaben und Rahmenbedingungen des Projekts umfassten die Nutzung von Python als Hauptprogrammiersprache, was eine effiziente Entwicklung und Datenverarbeitung ermöglichte. Für die grafische Benutzeroberfläche wurde Tkinter ausgewählt, eine Standardbibliothek in Python für GUI-Anwendungen, die eine intuitive und benutzerfreundliche Gestaltung erlaubt. Zudem war der Einsatz von pandas für die Datenverarbeitung und -analyse wesentlich, um die umfangreichen Datensätze von Open Powerlifting effektiv zu handhaben und zu vergleichen.

# Projektablauf

1. Ursprüngliche Projektplanung:

Ein Bild, das Text, Diagramm, Screenshot, Plan enthält.

Automatisch generierte Beschreibung

1. Ist-Analyse: In der Ist-Analyse wurde ein grundlegender Überblick über existierende Powerlifting-Tools gewonnen. Es wurde beobachtet, dass einige Anwendungen grundlegende Funktionen bieten, jedoch oft nicht speziell auf die Bedürfnisse von Powerlifting-Athleten zugeschnitten sind. Es fehlte an einer benutzerfreundlichen Schnittstelle, und viele Tools hatten keine umfassenden Vergleichsmöglichkeiten mit professionellen Daten. Diese Erkenntnisse führten zu dem Schluss, dass es Raum für ein einfacheres, zielgerichteteres Tool gibt.
2. Soll-Analyse: Das Ziel des Projekts war es, ein einfaches und intuitives Tool für Powerlifter zu entwickeln, das sich auf die Kernfunktionalitäten der Trainingsintensitätsberechnung und des Vergleichs mit Open Powerlifting-Daten konzentriert. Es sollte eine benutzerfreundliche Oberfläche bieten, die auch für weniger technisch versierte Nutzer leicht zugänglich ist. Das Tool sollte die Lücke in der Verfügbarkeit von spezialisierten, leicht bedienbaren Anwendungen für Powerlifting-Athleten schließen.Meilensteinplanung

# Realisierung

1. Der Aufbau des Systems für die RPE-basierte Trainingsintensitätsberechnung umfasst eine mehrschichtige Struktur, die sich auf die Trennung von Datenverarbeitung, Benutzeroberfläche und Logik konzentriert. Der Kern des Systems ist in Python geschrieben, wobei die Datenverarbeitung und die Geschäftslogik in separaten Modulen organisiert sind. Die Benutzeroberfläche wurde mit der Bibliothek CustomTkinter entwickelt, die eine intuitive und ansprechende Benutzerinteraktion ermöglicht. Die wichtigsten Dateien und Module des Systems sind:

main.py: Die Hauptdatei, die die Anwendung initialisiert und die GUI startet.

rpe\_percentages.py: Ein Modul, das die RPE-Logik und Berechnungen enthält.

openpowerliftingdata.csv: Eine CSV-Datei, die als Datenquelle für die Leistungsvergleiche dient.

1. Die Installation der Software für dieses Projekt war relativ unkompliziert und fokussierte sich auf die Grundanforderungen für ein Python-basiertes Programm:

**Python Installation**: Zuerst wurde Python installiert. Dies ist die Grundvoraussetzung, da die gesamte Anwendung in Python geschrieben ist. Die Installation erfolgte direkt über die offizielle Python-Website.

**Bibliotheken installieren**: Die für das Projekt erforderlichen Python-Bibliotheken, insbesondere Pandas für die Datenverarbeitung und CustomTkinter für die Gestaltung der Benutzeroberfläche, wurden mittels pip, dem Paketmanager von Python, installiert.

1. Die Konfiguration der Anwendung war ein wesentlicher Schritt, um die Funktionalität zu gewährleisten:

**Konfiguration der Dateipfade**: Die Pfade zu den erforderlichen Dateien, einschließlich der openpowerliftingdata.csv, wurden im Code definiert, um eine korrekte Datenverarbeitung zu ermöglichen.

**GUI-Einstellungen**: Die Benutzeroberfläche wurde konfiguriert, um eine benutzerfreundliche und intuitive Erfahrung zu bieten. Dies umfasste die Festlegung von Layout, Farbschema und Interaktionselementen.

**Testläufe und Debugging**: Vor der endgültigen Inbetriebnahme wurden umfangreiche Tests durchgeführt, um Fehler zu identifizieren und zu beheben.

Durch diese Schritte konnte eine voll funktionsfähige und benutzerfreundliche Anwendung realisiert werden, die speziell auf die Bedürfnisse von Powerliftern zugeschnitten ist.

# Qualitätssicherung

1. Die Qualität des entwickelten Tools wurde durch Benchmark-Tests geprüft. Dabei lag der Fokus auf zwei Hauptfunktionen: der Berechnung der Trainingsintensität und dem Vergleich der Benutzerdaten mit den Open Powerlifting-Daten.
2. Ergebnisse:

**Berechnungsfunktion** Die Funktion zur Berechnung der Trainingsintensität lief in durchschnittlich 0.000087 Sekunden (87 Mikrosekunden) ab. Dies zeigt, dass die Funktion sehr schnell arbeitet.

**Vergleichsfunktion**: Die Funktion zum Vergleich mit den Open Powerlifting-Daten benötigte im Durchschnitt 0.0153 Sekunden (15.3 Millisekunden). Dies ist etwas langsamer als die Berechnungsfunktion, aber immer noch schnell genug, um für die Benutzer effektiv zu sein.

Diese Ergebnisse zeigen, dass das Tool gut funktioniert und schnelle Antworten liefert, was für eine gute Benutzererfahrung wichtig ist.

# Projektabschluss

1. Das Projekt zielte darauf ab, eine benutzerfreundliche Anwendung zur Berechnung von RPE-basierten Trainingsintensitäten für Powerlifter zu entwickeln. Durch die Integration einer intuitiven Benutzeroberfläche und die effiziente Verarbeitung von Leistungsdaten konnte eine Anwendung geschaffen werden, die den Anwendern wertvolle Einblicke in ihre Trainingsfortschritte bietet.
2. Das Projektziel wurde weitgehend erreicht. Die Anwendung ermöglicht es Benutzern, ihre Trainingsintensitäten auf Basis von RPE-Werten zu berechnen und sich mit einer umfangreichen Datenbasis zu vergleichen. Die Benutzeroberfläche ist klar und benutzerfreundlich gestaltet, und die Anwendung liefert zuverlässige und nützliche Ergebnisse.
3. Während des Projekts stellten sich einige Herausforderungen, vor allem in Bezug auf die Datenverarbeitung. Eine signifikante Herausforderung war der Umgang mit großen Datensätzen, die zu Verzögerungen beim Start der Anwendung führten. Eine zusätzliche Herausforderung bestand darin, dass für die Entwicklung der Anwendung Python zum ersten Mal eingesetzt wurde. Dies führte zu einer steilen Lernkurve, da die Sprache und ihre spezifischen Bibliotheken neu erlernt werden mussten. Trotz dieser Anfangsschwierigkeiten konnte durch kontinuierliches Lernen und Anpassen der Methoden eine effiziente Lösung gefunden werden. Diese Erfahrung hat nicht nur das technische Know-how erweitert, sondern auch wichtige Fähigkeiten im Bereich des schnellen Lernens und der Anpassung an neue Technologien gefördert.
4. Es gibt verschiedene Bereiche, in denen die Anwendung weiter optimiert werden könnte:

**Lazy Loading**: Verzögertes Laden von Daten, um die Startzeit der Anwendung zu reduzieren.

**Separater Datenverarbeitungsthread**: Um die GUI reaktionsfähig zu halten, während Daten im Hintergrund geladen und verarbeitet werden.

**Vorab-Filterung von Daten**: Reduzierung der zu ladenden Datenmenge durch gezieltes Filtern.

**Effizientes Datenmanagement**: Weitere Optimierung der Datenverarbeitungsmethoden, insbesondere in Bezug auf Pandas-Operationen.

**Daten-Caching**: Zwischenspeichern von verarbeiteten Daten, um bei wiederholtem Start der Anwendung Zeit zu sparen.

**Verwendung einer Datenbank**: Für eine effizientere Datenverwaltung und -abfrage, insbesondere bei sehr großen Datensätzen.

# Dokumentation (Anhänge)

README Instruktionen sind im RPE-Calculator Ordner vorhanden.

# Quellenangaben

**CustomTkinter (Python Library)**:

* CustomTkinter Development Team. (2023). CustomTkinter: An enhanced version of the Tkinter library.   
  Verfügbar unter:   
  https://github.com/TomSchimansky/CustomTkinter

**Online-Ressourcen und Dokumentationen**:

* Python Software Foundation. (2023). Python 3.9.1 Dokumentation.   
  Verfügbar unter:   
  https://docs.python.org/3/

**Open-Source-Software und Bibliotheken**:

* Pandas Development Team. (2023). Pandas Dokumentation.   
  Verfügbar unter:   
  https://pandas.pydata.org/pandas-docs/stable/

**Datenquellen**:

* Open Powerlifting. (2023). Powerlifting Datensatz.   
  Verfügbar unter:   
  https://openpowerlifting.gitlab.io/opl-csv/