

# ΤΕΧΝΙΚΗ ΑΝΑΦΟΡΑ ΕΡΓΑΣΙΑΣ: "FITNESS ASSISTANT"

**ΟΝΟΜΑΤΕΠΩΝΥΜΟ:** Αρτουρ Γκαλστιαν, Τζιμας Χρηστος, Στελιος Σουλιотης **ΑΡΙΘΜΟΣ**  
**ΜΗΤΡΩΟΥ:** it22020, it2024092, it2024108 **ΜΑΘΗΜΑ:** Αντικειμενοστρεφής  
Προγραμματισμός (Java)

## 1. Εισαγωγή

Η εφαρμογή **Fitness Assistant** είναι ένα εργαλείο γραμμής εντολών (CLI) για την ανάλυση αρχείων προπόνησης τύπου **.tcx** (Training Center XML). Η εφαρμογή αναλύει δεδομένα από διάφορα αθλήματα (Τρέξιμο, Κολύμβηση, Ποδηλασία κ.α.), υπολογίζει στατιστικά (απόσταση, χρόνο, θερμίδες, ρυθμό) και τα παρουσιάζει στον χρήστη.

Η υλοποίηση βασίστηκε στις αρχές του Αντικειμενοστρεφούς Προγραμματισμού, με έμφαση στην επεκτασιμότητα και την ακρίβεια των υπολογισμών ακόμα και σε περιπτώσεις ελλιπών δεδομένων GPS.

## 2. Οδηγίες Μεταγλώττισης & Εκτέλεσης (Maven)

Η εφαρμογή χρησιμοποιεί το σύστημα **Apache Maven** για τη διαχείριση εξαρτήσεων και τη διαδικασία build.

### 2.1 Μεταγλώττιση (Build)

Για τη δημιουργία του εκτελέσιμου αρχείου **.jar**, ανοίξτε ένα τερματικό στον ριζικό φάκελο του έργου (εκεί που βρίσκεται το **pom.xml**) και εκτελέστε:

```
Bash  
mvn clean package
```

Η διαδικασία θα δημιουργήσει τον φάκελο **target**, μέσα στον οποίο θα βρίσκεται το αρχείο **FitnessAssistant-1.0-SNAPSHOT.jar**.

## 2.2 Εκτέλεση

Η εφαρμογή δέχεται ορίσματα (arguments) για τα χαρακτηριστικά του χρήστη και τα αρχεία προς ανάλυση.

### Σύνταξη:

Bash

```
java -jar target/FitnessAssistant-1.0-SNAPSHOT.jar [FLAGS] [FILES]
```

### Παράδειγμα Χρήσης:

Bash

```
java -jar target/FitnessAssistant-1.0-SNAPSHOT.jar -w 80 -g Male -a 30 "running.tcx" "swim.tcx"
```

### Επεξήγηση Ορισμάτων:

- `-w [kg]`: Βάρος χρήστη (Δεκαδικός, π.χ. 80.5). Απαραίτητο για τον υπολογισμό θερμίδων.
- `-g [Male/Female]`: Φύλο χρήστη. Επηρεάζει τον ακριβή τύπο θερμίδων (Heart Rate based).
- `-a [years]`: Ηλικία χρήστη.
- `[files]`: Λίστα από μονοπάτια αρχείων .tcx (διαχωρισμένα με κενό).

## 3. Αρχιτεκτονική & Σχεδιαστικά Πρότυπα (Design Patterns)

Η σχεδίαση του κώδικα ακολουθεί αυστηρά την ιεραρχία των δεδομένων TCX (Activity > Lap > Track > TrackPoint).

### 3.1 Factory Pattern (ActivityFactory)

Για τη δημιουργία των αντικειμένων δραστηριότητας, χρησιμοποιήθηκε το **Factory Design Pattern**. Η κλάση `ActivityFactory` διαβάζει το πεδίο `Sport` από το XML και επιστρέφει δυναμικά το κατάλληλο στιγμιότυπο:

- `RunningActivity` (για "Running")

- `CyclingActivity` (για "Biking" ή "Cycling")
- `SwimmingActivity` (για "Swimming")
- `WalkingActivity` (για "Walking")
- `DefaultActivity` (για "Other" ή άγνωστα αθλήματα)

Αυτό επιτρέπει την εύκολη προσθήκη νέων αθλημάτων στο μέλλον χωρίς αλλαγές στον κώδικα του Parser (`TcxLoader`).

### 3.2 Κληρονομικότητα & Πολυμορφισμός

- **Interface Activity:** Ορίζει το συμβόλαιο για όλες τις δραστηριότητες.
- **Abstract/Base Class DefaultActivity:** Υλοποιεί την κοινή λογική (διαχείριση Laps, υπολογισμούς χρόνου).
- **Subclasses:** Οι ειδικές κλάσεις (π.χ. `RunningActivity`) κληρονομούν από την `DefaultActivity` και ορίζουν δικούς τους πολλαπλασιαστές (multipliers) για τις θερμίδες (METs).

## 4. Αλγόριθμοι & Σχεδιαστικές Αποφάσεις

### 4.1 πολογισμός Απόστασης (Track.java)

- Στο **Τρέξιμο/Ποδηλασία**, υπάρχουν δεδομένα GPS (Latitude, Longitude).
- Στην **Κολύμβηση (Πισίνα)** ή στον **Διάδρομο**, ΔΕΝ υπάρχουν δεδομένα GPS, αλλά το αρχείο περιέχει αθροιστική απόσταση (`DistanceMeters`).

Υλοποιήθηκε στην κλάση `Track` ένας **αλγόριθμος**:

Η μέθοδος υπολογίζει τη διαφορά (`delta`) του πεδίου `DistanceMeters` (Τελευταίο Σημείο - Πρώτο Σημείο).

### 4.2 Στρατηγική Υπολογισμού Θερμίδων

Ο υπολογισμός των θερμίδων γίνεται με δύο επίπεδα ακρίβειας:

1. **Προηγμένος Τύπος (Heart Rate Based):** Αν το αρχείο περιέχει καρδιακούς παλμούς ΚΑΙ ο χρήστης έχει δώσει φύλο/ηλικία/βάρος, χρησιμοποιείται ο τύπος:
  - a. *Male:*  $((-55.0969 + (0.6309 * HR) + (0.1966 * W) + (0.2017 * A)) / 4.184) * Time$
  - b. *Female:*  $((-20.4022 + (0.4472 * HR) + (0.1263 * W) + (0.074 * A)) / 4.184) * Time$

2. **Απλός Τύπος (METs):** Αν λείπουν δεδομένα, χρησιμοποιείται ο τύπος  $MET * Weight * Hours$ , όπου το MET (Multiplier) ορίζεται διαφορετικά σε κάθε υποκλάση (π.χ. Running=10, Cycling=8).

### 4.3 Pace vs Speed

Η εφαρμογή προσαρμόζει την έξοδο ανάλογα με το άθλημα:

- Για το **Τρέξιμο**, εμφανίζεται **Μέσος Ρυθμός (Pace)** σε min/km (υπολογισμένο ως  $Time / Distance$ ).
- Για τα υπόλοιπα αθλήματα, εμφανίζεται **Μέση Ταχύτητα** σε km/h.

## 5. Δομή Δεδομένων

Η ανάγνωση του αρχείου XML γίνεται μέσω DOM Parser (TcxLoader). Τα δεδομένα χαρτογραφούνται στη μνήμη ως εξής:

- Main -> List<Activity>
- Activity -> List<Lap>
- Lap -> List<Track>
- Track -> List<TrackPoint>

Κάθε TrackPoint αποθηκεύει: Timestamp, Position (Lat/Lon), Altitude, Distance, HeartRate, Speed, Cadence.