

# Projet : minuteur HIIT

Training : <http://developer.android.com/training/>

Guide : <http://developer.android.com/guide/>

Documentation : <http://developer.android.com/reference/>

Documentation java : <http://docs.oracle.com/javase/8/docs/api>

Tutoriel en français : <http://fr.openclassrooms.com/informatique/cours/creez-des-applications-pour-android>

## Présentation

L'objectif de ce projet est de développer un minuteur HIIT pour smartphone ou tablette android. Le HIIT (High Intensity Interval Training) ou entraînement fractionné de haute intensité est un mode d'entraînement fractionné qui vise un renforcement de la condition physique par de brèves séances (de 15 à 30 minutes) d'exercices en anaérobie ([source wikipédia](#)). Le minuteur que vous réaliserez permettra à un sportif de programmer ses séquences d'entraînement HIIT, par exemple : 5 fois 1 minute de travail (pompes, course, etc.) et 30 secondes de repos, suivi d'un repos long de 3 minutes et à nouveau 5 fois 1 minute à de travail (pompes, course, etc.) et 30 secondes de repos.

Le schéma explicatif ci-dessous décrit le principe d'une séance d'entrainement fractionné que votre application devra suivre. Vous devrez respecter le même vocabulaire : **préparation, séquence, cycle, travail, repos, repos long**.

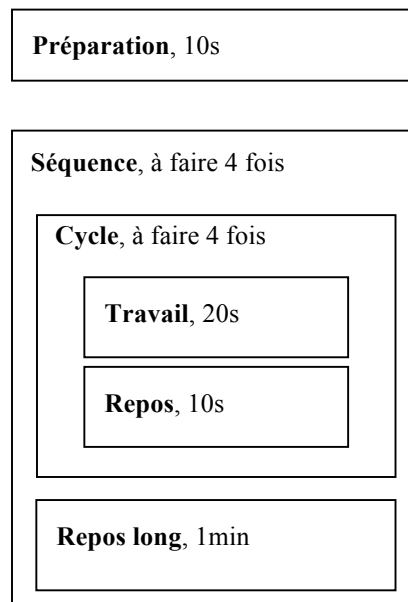


Figure 1 : Exemple d'une séance d'entrainement fractionné

Les fonctionnalités de base de votre application seront :

- Créer une séance d'entraînement;
- « Jouer » une séance d'entraînement (avec des indications de couleurs et sonore) ;
- Sauvegarder une séance d'entraînement ;
- Lister les séances d'entraînements.

Une fois ces fonctionnalités développées, vous pourrez améliorer votre application avec de nouvelles fonctionnalités, par exemple :

- Conserver un calendrier des entraînements ;
- Associer des photos à votre entraînement ;
- Informer les réseaux sociaux de votre entraînement ;
- Enregistrer votre parcours à l'aide du GPS ;
- Enregistrement des entraînements sur serveur (firebase) ;
- Etc.

*Exemple d'applications sur le play store :*

- Tabata Timer : Interval Timer Workout Timer HIIT

<https://play.google.com/store/apps/details?id=com.evgeniysharafan.tabatatimer&hl=fr>

- Tabata Timer for HIIT

<https://play.google.com/store/apps/details?id=com.simplevision.workout.tabata&hl=fr>

#### **EVALUATION :**

Le projet se base sur la notion de « développer plus pour gagner plus » ! Une application minimale vous est demandée (ci-dessus et développé dans les étapes ci-après). Des fonctionnalités supplémentaires vous sont proposées pour avoir des points supplémentaires. Vous pourrez choisir vos fonctionnalités en fonction de vos envies.

Vous ferez attention à ce que votre code soit **correctement structuré**, en distinguant les parties modèle (classes java), vue (XML) et contrôleur (Activités). De plus, votre **code devra être clair et lisible** : correctement indenté, respecter les conventions de nommage et utiliser des noms explicites. Vous **commenterez les parties de code** un peu longue afin de détailler les étapes de vos algorithmes.

**AUCUN BUG** ne devra être présent dans votre application sous peine d'un fort malus !

### RENDU du projet :

Vous devrez rendre avec votre projet un rapport. Ce document, **SOUS FORME PDF**, expliquera les fonctionnalités présentes (minimales et supplémentaires) dans votre projet et le workflow de votre application.

Enfin, le **nom de votre projet devra commencer par votre login**, de même que l'archive que vous rendrez.

Le non respect de ces consignes impactera fortement l'évaluation de votre projet.

## Etape 1 : Mise en place et réalisation du workflow

**Objectif** : concevoir l'enchaînement des différentes parties de l'application sur papier et réaliser ce workflow en utilisant les intentions explicites vu dans le Cours2/TP2.

Mettez en place le workflow de votre application. Pour ce faire, réalisez un storyboard sur **papier** représentant les différentes fenêtres composant votre application puis réaliser les différentes fenêtres/activités de votre application.

## Etape 2 : créer une séance d'entraînement

**Objectif** : mettre en place une interaction utilisateur et une récupération des données (réf. cours 1 / TP1). Transfert de données entre activités (réf. cours 2 / TP 2).

Réalisez une interface permettant la création d'une séance d'entraînement : **préparation, séquence, cycle, travail, repos, repos long**. Lorsque l'utilisateur « jouera » la séance, vous devrez envoyer l'objet créé à l'activité suivante (étape 3).

**IMPORTANT** : vous **DEVEZ** séparer vos données, vos vues et votre traitement (MVC : modèle, vue, contrôleur). Par exemple, dans l'exercice pierre/feuille/ciseaux, les données étaient représentées par les classes `Jeu` et `Resultat`, la vue par la représentation XML de l'activité et le traitement par l'activité même.

## Etape 3 : « Jouer » une séance d'entraînement

**Objectif** : mettre en place un compte à rebours complexe (pour se faire vous utiliserez la classe `CountDownTimer`).

Vous devrez dans un premier temps vous familiariser avec la classe `CountDownTimer` puis prendre en compte la séance d'entraînement créée dans l'étape 1. L'utilisateur devra être prévenu des changements d'états par des effets sonores et visuels.

<https://developer.android.com/reference/android/os/CountDownTimer>

## Etape 4 : Sauvegarder et lister les séances d'entraînements

**Objectif** : manipulation de la base de données interne (réf. cours 3 / TP3).

## **Etape 5 : Respect du cycle de vie**

**Objectif** : comprendre et utiliser les mécanismes liés au cycle de vie d'une application (réf. cours 4).

## **Etape 6 : Développer plus pour gagner plus**

À venir...