

Rendu TP2 : Les capteurs

Meryam RHADI
Lamyae KHAIROUN

27 février 2022

Master 1
génie logiciel
UE Programmation mobile

Responsable
Abdelhak-Djamel Seriai
Bachar Rima

Sommaire

Titre	1
Sommaire	2
1 Introduction	3
2 hiérarchie du projet	3
3 Exercice 1 : Liste de capteurs	4
4 Exercice 2 : Détection de présence/absence de capteurs	5
5 Exercice 3 : Accéléromètre	6
6 Exercice 4 : Direction	7
7 Exercice 5 : Secouer un appareil	9
8 Exercice 6 : Proximité	9
9 Exercice 7 : Géolocalisation	10

1 Introduction

Ce tp à pour but de comprendre le fonctionnement et la programmation des capteurs sous android, en utilisant java.

2 hiérarchie du projet

Pour bien organiser notre travail, on a décidé de faire une activity : **MainActivity** commune à tous les exercices qui contiennent autant de boutons que d'exercices, et ce bouton à chaque fois lors de cliques il lance l'exercice correspondant.

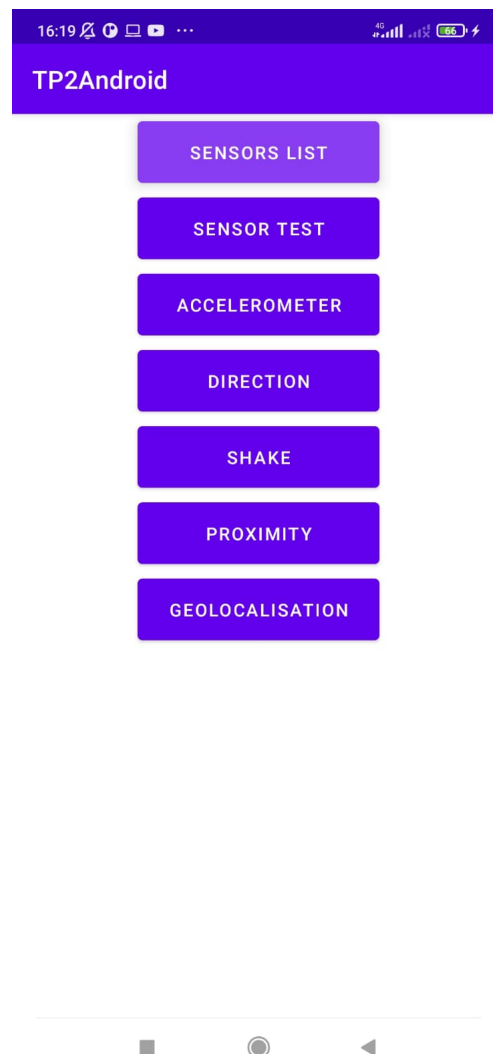


Figure 1. Page d'accueil

3 Exercice 1 : Liste de capteurs

Pour le première exercice, on a créé une application qui affiche tous les capteurs disponibles sur le téléphone ou on installe l'application, et pour ça on a utilisé **SENSOR_SERVICE** et **sensor.TYPE_ALL**, et on a affiché cette liste dans un **Toast**.

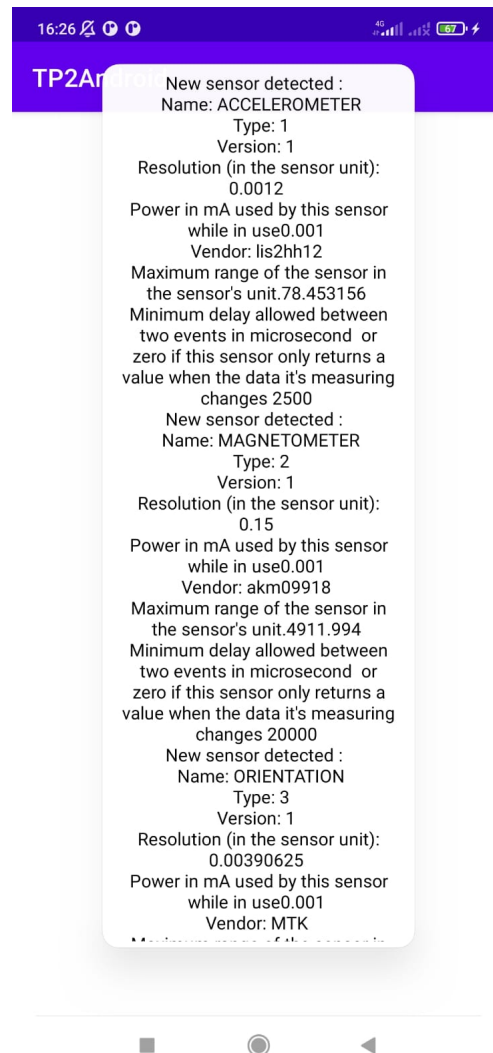


Figure 2. Liste des capteurs disponible

4 Exercice 2 : Détection de présence/absence de capteurs

Dans cet exercice, on a créé une application qui informe l'utilisateur de l'indisponibilité ou la disponibilité d'un capteur donné sur son appareil, pour ça on a testé sur les capteurs de la proximité et de magnétique, et puisque dans notre cas les deux ont été disponibles on avait un petit affichage qui confirme ça.



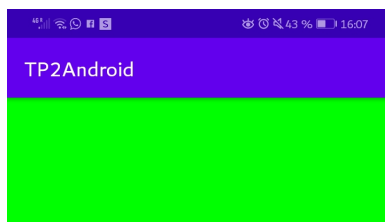
Figure 3. Capture d'écran affichant la disponibilité des capteurs

5 Exercice 3 : Accéléromètre

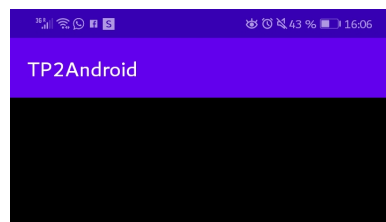
Dans cette exercice on a développé une application qui affiche des couleurs différentes comme fond d'écran en fonction des valeurs de l'accéléromètre. Tel que la couleur :

- Rouge pour les valeurs inférieures.
- Noir pour les valeurs moyennes.
- Vert pour les valeurs supérieures.

On a utiliser une fonction qui calcule l'accélération selon les trois axes : x, y et z.

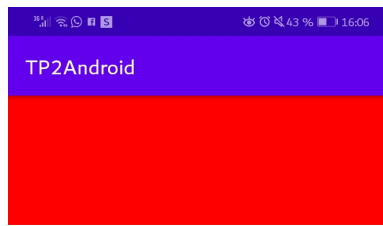


Acceleration: 20
Previous Accel: 20
Change value: 5



Acceleration: 11
Previous Accel: 11
Change value: 3





Acceleration: 10
Previous Accel: 10
Change value: 0



6 Exercice 4 : Direction

Cette application permet d'afficher sur l'écran la direction du mouvement : gauche, droite, haut, bas. Pour ce faire, on a utilisé le capteur Accéléromètre, et suivant les valeur de x et y on pouvait détecter la position du téléphone et la mouvement qu'il fait.



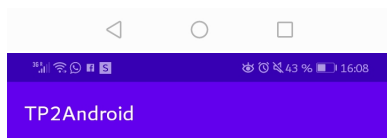
Haut
8.649465

Droit
0.24124908



Haut
7.472661

Gauche
-0.85710907



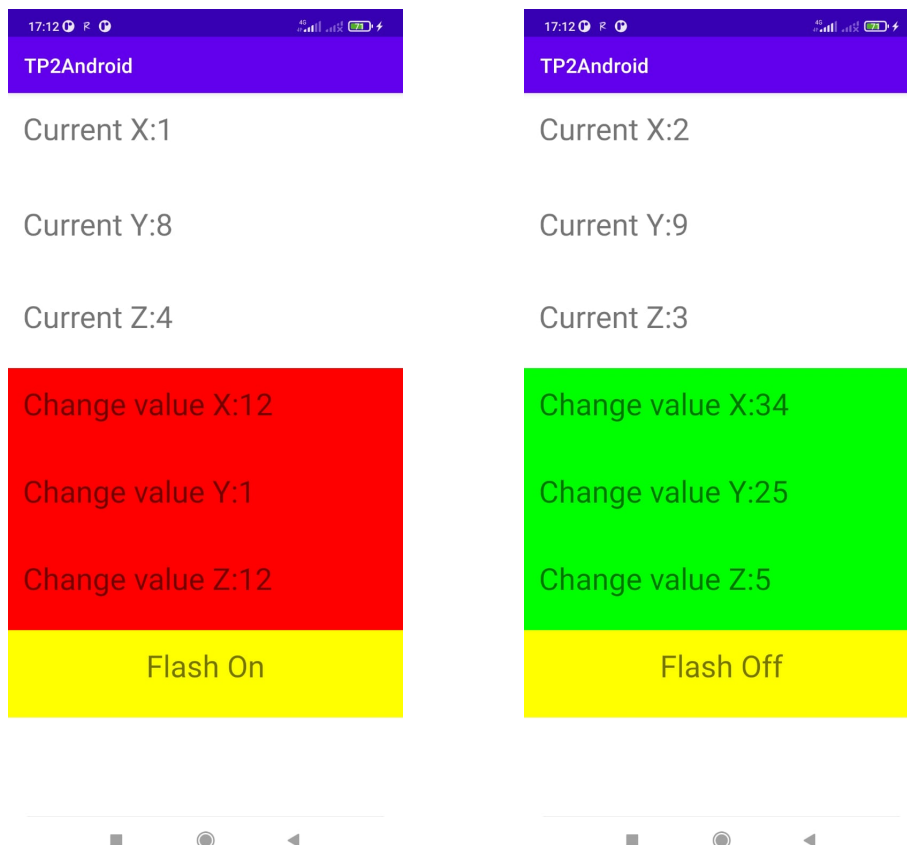
Bas
-9.473231

Droit
0.55506134



7 Exercice 5 : Secouer un appareil

Cette application permet d'allumer le flash en secouant l'appareil et l'éteint également en le secouant une deuxième fois. On a utilisé les capteurs de **CAMERA_SERVICE** avec la méthode **setTorchMode** pour allumer et éteindre le flash et **TYPE_ACCELEROMETER** pour détecter les valeurs de l'accéléromètre.



8 Exercice 6 : Proximité

Pour le sixième exercice on a défini une application qui en fonction de la proximité, affiche une image indiquant si l'objet est proche ou s'il est loin, pour cela on a utilisé **Sensor.TYPE_PROXIMITY** et un petit traitement qui permet de changer l'image en fonction de la proximité dans la méthode **onSensorChanged**.

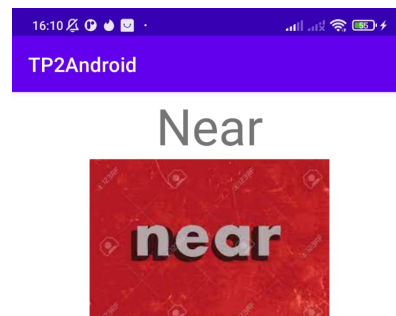
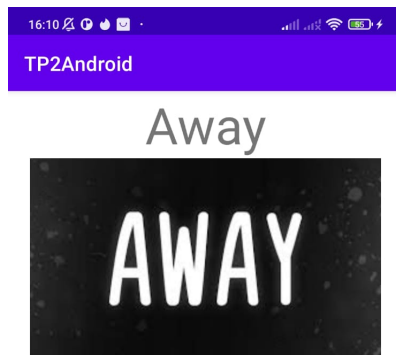


Figure 4. Capture d'écran affichant l'image dans l'état d'être loin du capteur de proximity

Figure 5. Capture d'écran affichant le changement d'état vers proche

9 Exercice 7 : Géolocalisation

Cette application demande d'afficher la position géographique (géolocalisation), de l'utilisateur de l'application. Pour ce faire on a utilisé le service **LOCATION_SERVICE**.

Pour utiliser l'application, il faut donner la permission pour utiliser la localisation dans le téléphone où l'application est installée.

Rendu TP2 : Les capteurs

17:26		4G		73	
Latitude	0.0	Latitude	43.610566		
Longitude	0.0	Longitude	3.8618582		
Altitude	0.0	Altitude	102.0		
CHOOSE SOURCE	GET POSITION	DISPLAY ADDRESS	CHOOSE SOURCE	GET POSITION	DISPLAY ADDRESS

Rendu TP2 : Les capteurs

17:27

4G

73

Latitude

43.610566

Longitude

3.8618582

Altitude

102.0

CHOOSE
SOURCE

GET
POSITION

DISPLAY
ADRESS

19 B Rue Gustave, 34000 Montpellier, France, 34000
Montpellier