

Application N°5.2_détecteur de choc communicant

Objectif

- Réaliser un détecteur de choc avec son Smartphone + envoi d'un SMS avec géolocalisation sur Google Maps

Mise en situation :

- *Prototypage d'un système de surveillance d'un colis fragile durant son transit*
- « Ange gardien » lors d'activités sportives (roller, vélo, ...)

Prérequis

- Application N°0 _ Détecteur de choc
- Application N°5 _ Gestion des SMS

Configuration

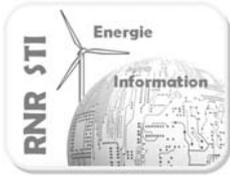
- Identique aux applications N° 0 et N°5

Remarque

Une procédure « cor_virgule » non détaillée ici corrige le format des données car les coordonnées GPS récupérées comportent une virgule au lieu d'un point (problème générique sur Android).

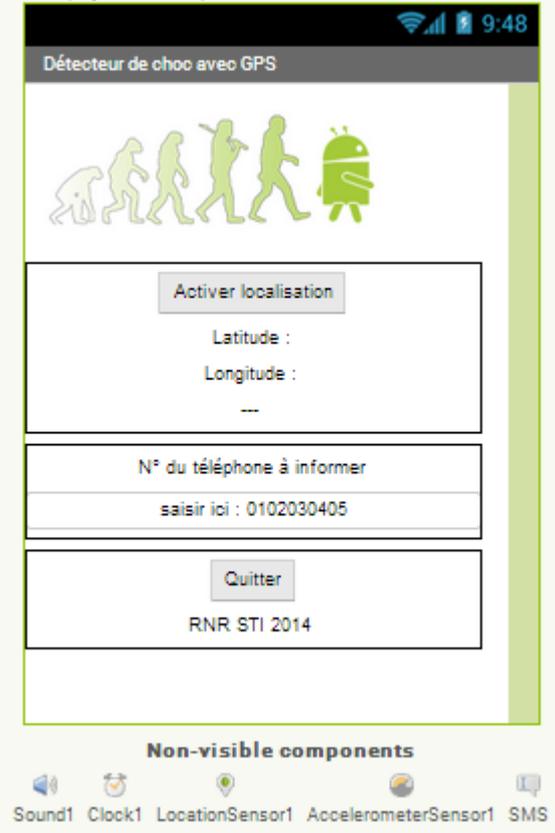
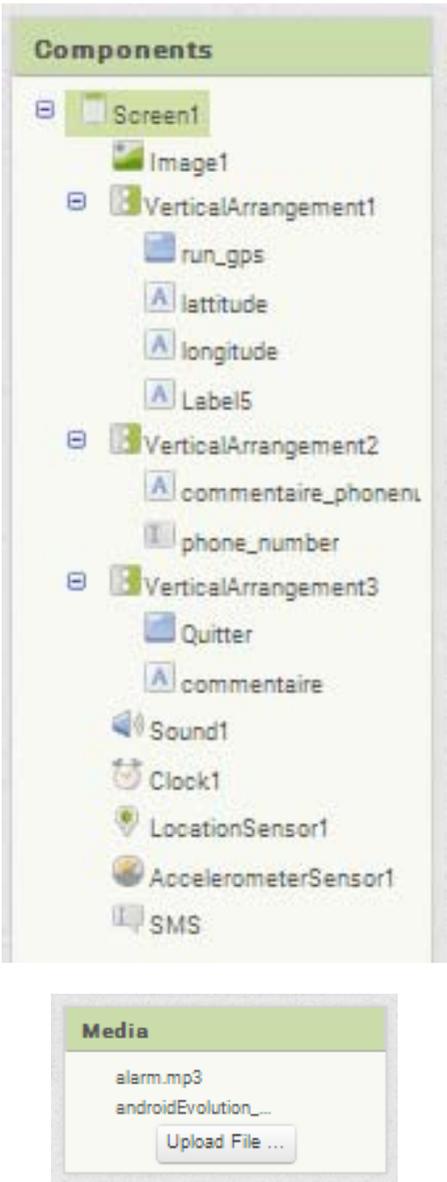


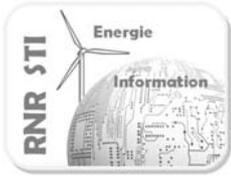
Dans Google Maps, choisissez votre position actuelle pour la destination (point B de l'itinéraire).



Application N°5.2_détecteur de choc communicant

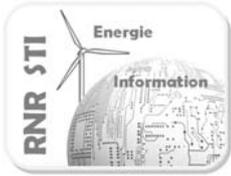
Application App Inventor du Smartphone

Ecran du terminal	Composants installés
	



Application N°5.2_détecteur de choc communicant

Programme de l'application ANDROID	Commentaires
	<p>Activatation de la géolocalisation</p> <p>Affichage de la latitude et la longitude</p> <p>Événement : choc détecté sur un des 3 axes</p> <p>Paramétrage du N° de téléphone cible du SMS</p> <p>Chainage des textes constituant le SMS dont l'adresse Google Maps avec les coordonnées lors du choc</p> <p>Emetre un son d'alarme</p> <p>Envoyer le SMS</p>



Application N°5.2_détecteur de choc communicant

Procédure de correction des valeurs numériques

```
to cor_virgule value
  result ← join
  segment text ← get value
  start ← 1
  length ← starts at text ← get value - | 1
  piece ← " "
  -
  segment text ← get value
  start ← 1 + starts at text ← get value
  piece ← "0"
  length ← length ← get value - starts at text ← get value
  piece ← "0"
```

Appel de la procédure

```
call cor_virgule
value ← LocationSensor1 . Longitude
```