

DEVOIR FINAL

Classe : SEM31

Matière : *Développement Mobile Avancé*

Nb pages : 4

Enseignant : *Souissi Hafedh*

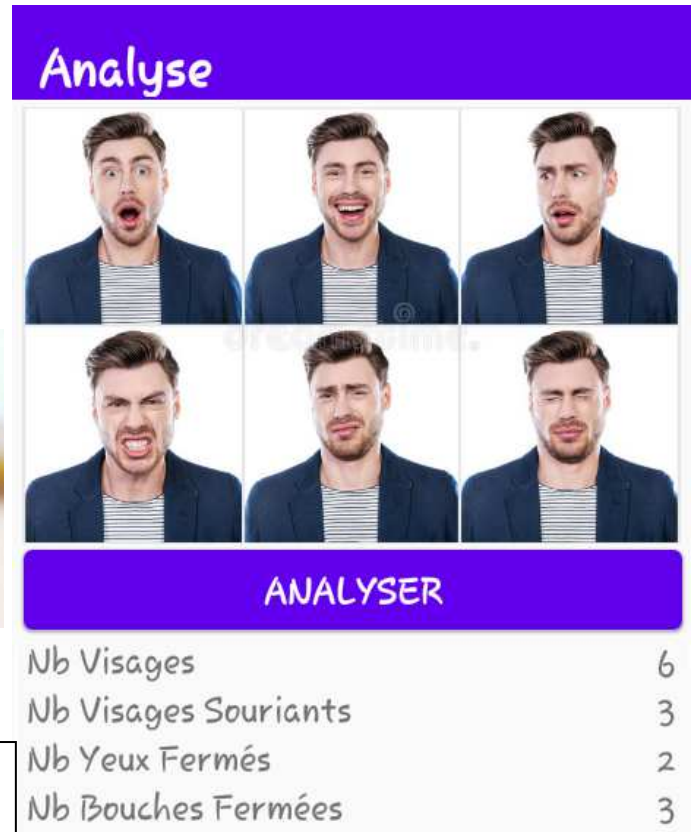
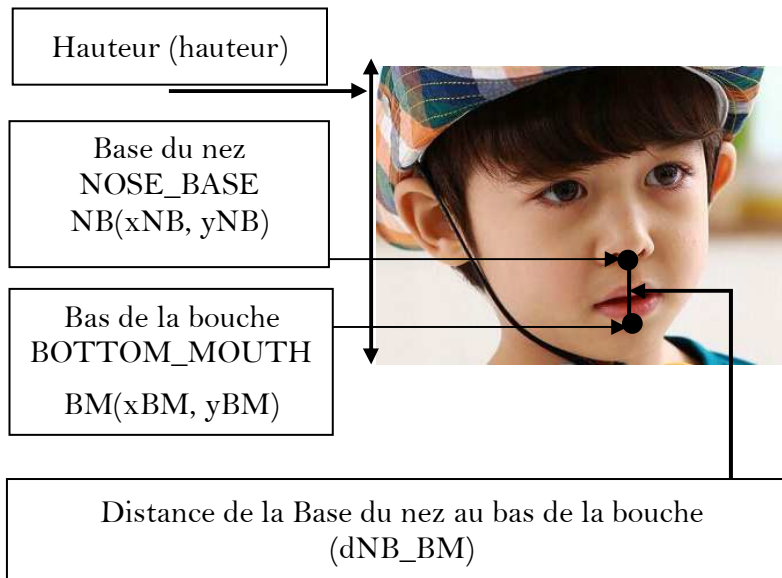
Documents Non Autorisés

Barème : 20 = 5 + 8 + 7

Durée : 1 heure 30 minutes

Exercice1 (Utiliser Annexe I)

"Analyse" est une application Android qui permet de détecter les visages d'une image et d'afficher un ensemble de statistiques sur les visages détectés. "Analyse" contient une seule activité.



Pour obtenir la distance "dNB_BM" entre les deux points NB(xNB, yNB) et BM(xBM, yBM), utiliser la formule : $dNB_BM = \sqrt{\text{Math.pow}(xNB - xBM, 2) + \text{Math.pow}(yNB - yBM, 2)}$.

Le coefficient $cNBMBH = dNB_BM / \text{hauteur}$ permet de savoir si la bouche est fermée ou non : Si $cNBMBH < 0,27$ alors la bouche est fermée.

La méthode "analyser()" permet de détecter les visages de l'image, de calculer et d'afficher :

- le nombre de visages,
- le nombre de visage souriant (la probabilité qu'il rit est strictement supérieure à 0.6),
- le nombre d'yeux droites et gauches fermés (la probabilité que l'œil est ouverte est strictement inférieur à 0.5),
- le nombre de bouches fermées.

1- Donner le code de la méthode "analyser()" de la classe "MainActivity".

Exercice2 (Utiliser Annexe II)

L'objectif de cet exercice est de réaliser une application android qui s'utilise dans un bateau pour connaître l'état de la mer en fonction des accélérations appliquées sur le bateau par les vagues et la pesanteur. L'état de la mer peut être "Mer Calme", "Mer Agitée" ou "Mer Très Agitée".

Le smartphone qui contient l'application est placé dans le bateau. C'est la norme des accélérations sur les trois axes (x, y et z) qui permet de déduire l'état de la mer. La norme de l'accélération est calculée par la formule : $\text{normeAcc} = \sqrt{\text{Math.pow}(x,2) + \text{Math.pow}(y,2) + \text{Math.pow}(z,2)}$

Le tableau suivant présente l'état de la mer et l'image à afficher en fonction de la norme des accélérations :

Norme des accélérations (normeAcc)	Etat	Image à afficher	Id de l'image à afficher
$\text{normeAcc} < 13$	Mer Calme		R.drawable.mer_calme
$13 \leq \text{normeAcc} < 17$	Mer Agitée		R.drawable.mer_agitee
$17 \leq \text{normeAcc}$	Mer Très Agitée		R.drawable.mer_tres_agitee

"EtatMer" est une application Android qui permet d'afficher la norme des accélérations interceptées par le capteur d'accélération et de modifier l'image affichée suivant la valeur interceptée en respectant le tableau précédent.

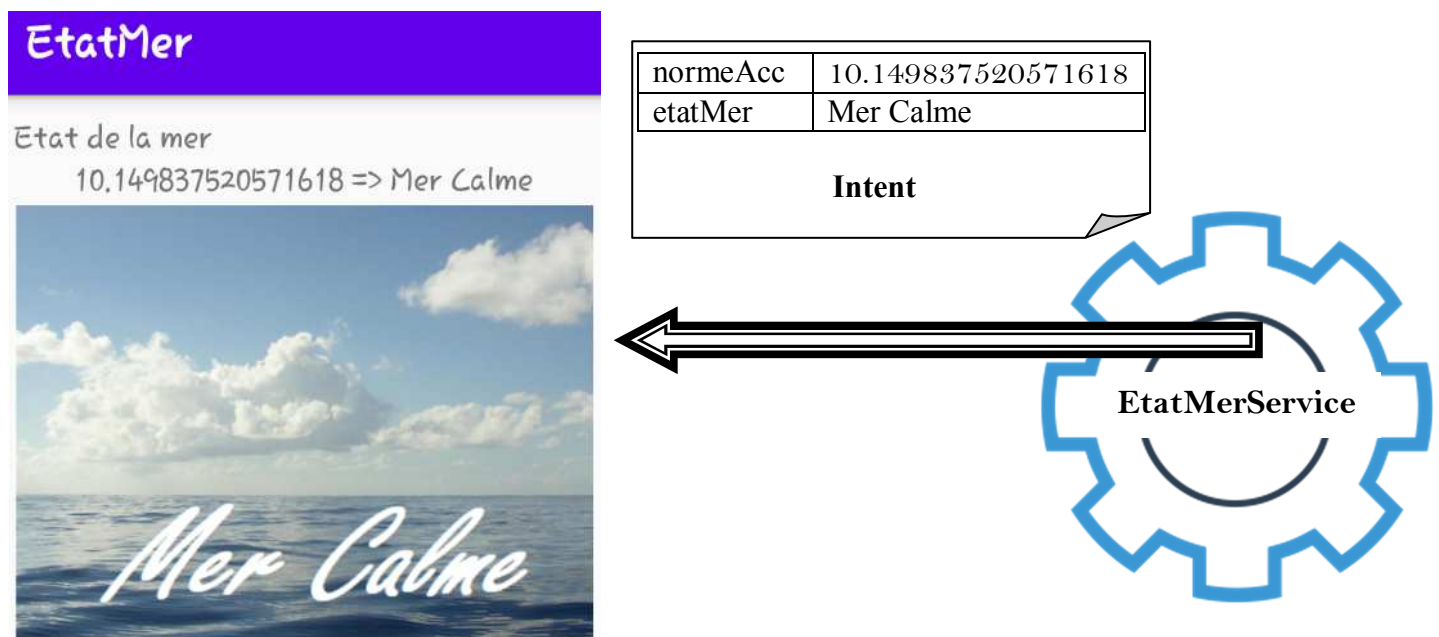
"EtatMer" est composée d'une activité "MainActivity" et d'un service "EtatMerService". "EtatMerService" utilise un capteur de type TYPE_ACCELEROMETER pour intercepter les

modifications des accélérations sur les trois axes, par la suite il calcule la norme des accélérations et déduit l'état de la mer et diffuse (broadcast) ses deux valeurs dans un Intent.

Lorsque "MainActivity" fait un "Resume" elle démarre le service et lorsqu'elle fait une "Pause" elle arrête le service. "MainActivity" utilise un BroadcastReceiver pour lire la valeur "normeAcc" et "etatMer" envoyée par le service et de les afficher dans "tvEtat". Suivant la valeur de "normeAcc", "imgEtat" affiche IMG_MER_CALME ou IMG_MER_AGITEE ou IMG_MER_TRES_AGITEE.

Le texte affiché dans "tvEtat" est de la forme suivante : normeAcc => Etat, par exemple pour une normeAcc = 10.149837520571618 le texte affiché est "10.149837520571618 => Mer Calme".

Ci-dessous, un exemple de fonctionnement de l'application :



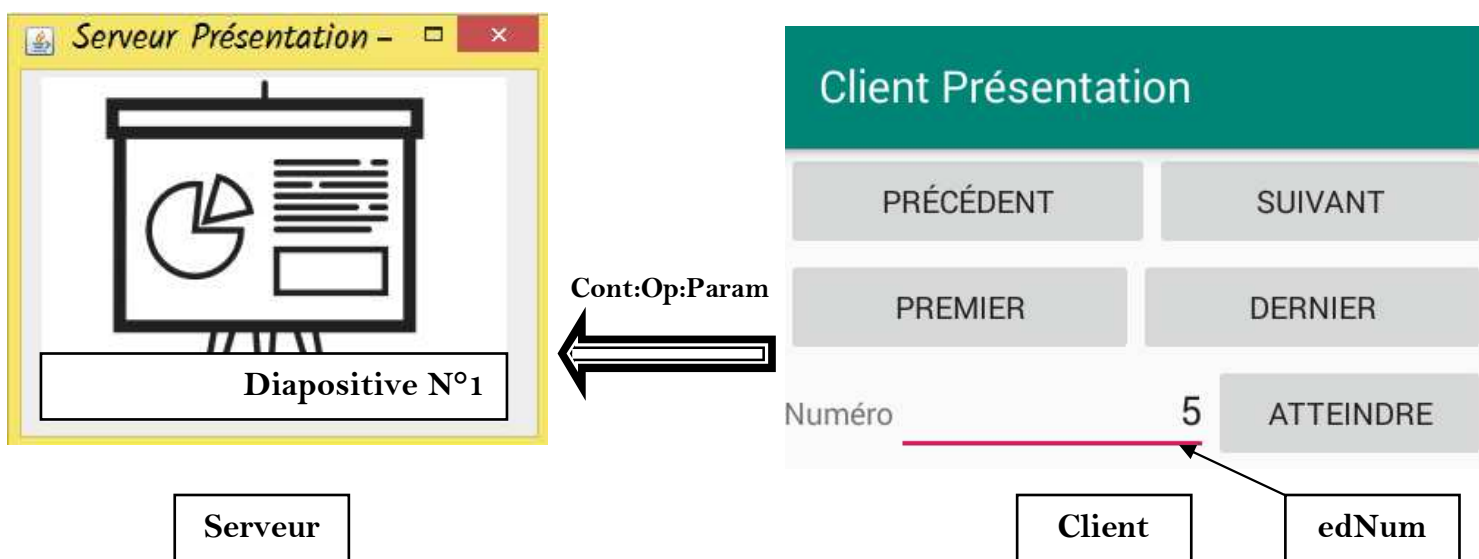
- 1- Donner le code des méthodes "ajouterEcouteur()", "demarrerService()", "arreterService()", "onResume()", "onPause()" et "actualiser(...)" de la classe "MainActivity".
- 2- Donner le code des méthodes "init()", "onDestroy()" et de "onSensorChanged()" de la classe "EtatMerService".

Exercice3 (Utiliser Annexe III)

On désire développer une application qui permet de contrôler une présentation composée de plusieurs diapositives par un smartphone. Cette application utilise les sockets. Le programme serveur est installé sur un ordinateur d'adresse "192.46.10.20". Le programme client est installé sur un Smartphone android pour envoyer les commandes de l'utilisateur au serveur. Les contrôles qu'on peut effectuer sur la présentation et les messages envoyés par le socket clients sont résumés dans le tableau suivant :

Contrôle (Cont)	Opération (Op)	Paramètre (Param)	Message envoyé
PrecSuiv (0)	Précédent (0)		0:0
	Suivant (1)		0:1
PremDern(1)	Premier(0)		1:0
	Dernier (1)		1:1
Att (2)	Atteindre (0)	numero	2:0:numero

Le fonctionnement de l'application est comme suit :



- 1-Donner le code des méthodes " lancerThreadServeur()", "demarrerServeur()" et "executerCommande(...)" de la classe "**ServPresentation**" (coté serveur).
- 2-Donner le code des méthodes " lancerThreadClient()", "demarrerClient()", "envoyer() ", "precedent() ", "suivant() ", "premier() ", "dernier()" et "atteindre()" de la classe "**MainActivity**" (coté client).