

**420-V41-SF**  
**PROGRAMMATION D'APPLICATIONS MOBILES I**  
**ÉVALUATION FINALE À CARACTÈRE SYNTHÈSE – PARTIE 1**  
**PONDÉRATION : 10%**

### OBJECTIFS PÉDAGOGIQUES

Ce travail vise à familiariser l'étudiante ou l'étudiant avec les objectifs suivants :

- 016T- Appliquer une approche de développement par objets.
- 017D- Concevoir et développer une application hypermédia dans des réseaux internes et mondiaux.

De même, il permet à l'étudiante ou à l'étudiant d'appliquer les standards de conception, de documentation et de programmation.

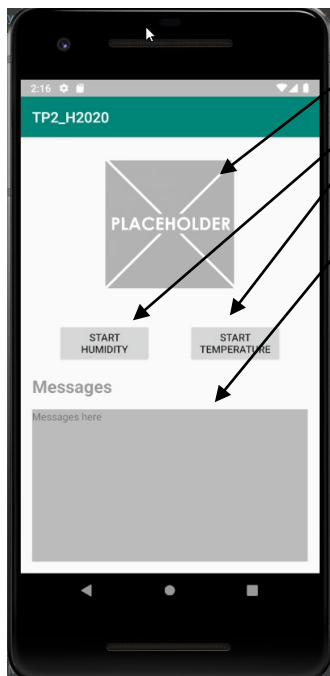
### MISE EN CONTEXTE ET MANDAT

Pour la première partie de ce dernier travail, vous devez réaliser une activité Android qui permet d'accéder à des données de température et d'humidité stockées sur un service Web.

Les données que vous devez acquérir seront utilisées dans les parties 2 et 3.

### CONTRAINTES DE RÉALISATION

Pour la partie 1 du travail, vous devez produire une application Android offrant les fonctionnalités suivantes :



- L'application doit afficher un logo de votre choix.
- L'application doit permettre d'aller chercher des données de température et d'humidité.
- L'application doit indiquer dans la zone prévue à cet effet tous les messages pertinents en lien avec l'acquisition des données. Cela inclut :
  - Un message indiquant le début de l'acquisition des données ainsi que l'adresse du service Web. Une barre de progression doit s'afficher pendant l'acquisition.
  - Un message indiquant le succès ou l'échec de l'acquisition.
  - Un message indiquant le nombre de données récupérées lorsque l'acquisition est un succès.

- Les données doivent provenir du serveur :

<https://v41-mobile-api.herokuapp.com/sensordata/>

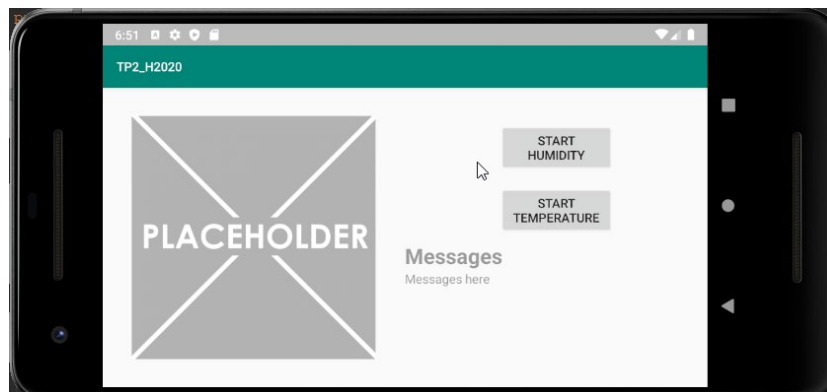
Les données de température correspondent au capteur #1,

<https://v41-mobile-api.herokuapp.com/sensordata/1>

tandis que les données d'humidité correspondent au capteur #2

<https://v41-mobile-api.herokuapp.com/sensordata/2>

- Il doit être possible de démarrer l'activité principale en lui envoyant l'adresse du serveur sous la forme d'une chaîne de caractères (lancement implicite de l'activité)
- L'activité principale doit supporter les changements d'orientation :



- Une fois les données acquises, l'activité principale doit afficher une autre activité (une pour l'humidité et une pour la température) et lui acheminer les données acquises (lancement explicite des activités). Les données doivent être envoyées en utilisant les « *parcels* ».
- Chaque nouvelle activité (température et humidité) doit alors afficher un message indiquant combien de données ont été reçues. Ces données seront utilisées dans les parties 2 et 3 du travail. On veut juste s'assurer qu'elles sont correctement transmises.
- Vous devez commenter le code en mode javadoc. La forme est ce qui est le plus important, mais le fond doit décrire avec justesse ce que vous faites et ce, **en français**. Vous devrez générer et remettre votre javadoc.

L'opérationnalité des fonctionnalités ne garantit pas la cote de passage. L'architecture et la qualité du code réalisé ainsi que des documents complet et professionnel sont primordiaux. Si le professeur le juge à propos, toute étudiante ou tout étudiant pourra être convoqué à une rencontre d'évaluation sur Teams pour vérifier son degré d'acquisition des connaissances et d'appréhension de la solution proposée.

## CONTEXTE DE RÉALISATION ET DÉMARCHE DE DÉVELOPPEMENT

Ce travail pratique doit être réalisé **seul**.

Les pages qui suivent résument les étapes du projet ainsi que les biens livrables devant être remis à la fin de chacune d'elles.

### Étape 1: Analyse de la situation

Dans cette étape, vous devez prendre connaissance du mandat qui vous est confié.

#### Biens livrables :

- Aucun.

#### Méthode:

- s/o

### Étape 2: Programmation de l'application

Vous devez procéder à la codification des fonctionnalités demandées.

#### Biens livrables :

- Code source de l'application documenté (javadoc). Vous devez aussi générer la javadoc dans un dossier à part.

#### Méthode:

- Vous devez remettre votre code source dans une archive .zip identifiée en fonction de votre nom. Par exemple, pour l'étudiant nommé Pierre Poulin, l'archive devrait être nommée TP1\_PierrePoulin.zip.
- Vous devez aussi remettre le fichier RemiseTP1.doc accompagné de la grille d'auto-évaluation correctement complétée.

Cette archive doit être déposée sur LÉA **avant le 27 avril 2020 à 7h59.**

### MODALITÉS D'ÉVALUATION

Vous trouverez dans le fichier RemiseEFCS - Partie 1.docx la grille de correction qui sera utilisée pour le travail. **Cette grille indique la pondération accordée à chacune des parties du projet.**

- Tous les **biens livrables** de chacune des étapes devront être **remis à temps** et selon les modalités spécifiées.