

Énoncé du Travail Pratique 3

4 décembre 2018

Préparé par
Benjamin Lemelin

1 Résumé

Développer une application permettant de chiffrer et de déchiffrer des messages. La clé de chiffrement devra être obtenue en appelant un serveur dédié.

2 Conditions de réalisation

Valeur de la note finale	Type	Durée	Nombre de remises
25 %	En équipe de 2	3 semaines (Excluant les congés)	1

3 Spécifications

3.1 Projet de départ

Un projet de départ est fourni avec cet énoncé. Si vous le placez dans [un dépôt Git](#), vous devez en donner accès à votre professeur le plus tôt possible.

Le code fourni contient pratiquement tout ce dont vous aurez besoin pour créer le projet, soit :

- Les [activités](#) (Incomplet)
- Un [dialogue](#) pour demander un nombre (nommé « KeyPickerDialog ») (Complet)
- Un bout de code pour [placer du texte dans le presse-papier](#) (voir « MainActivity ») (Complet)
- Un [filtre](#) pour filtrer [l'entrée d'un « EditText »](#) (nommé « CharactersFilter ») (Complet)
- Les fichiers XML d'interface utilisateur (Complet)
- Les chaînes de caractères (fichier « [strings.xml](#) ») (Incomplet)
- Les [images vectorielles](#) de l'interface ainsi que [l'icône adaptative](#) (Complet)
- Le fichier « [AndroidManifest.xml](#) » (Incomplet)
- Les [fichiers « Gradle »](#) avec les dépendances nécessaires au projet (Complet)

Prenez le temps de vous familiariser avec le code existant. **Vous risquez de perdre énormément de temps si vous ne le faites pas!** Pour vous aider, consultez la documentation incluse.

Familiarisez-vous aussi avec l'interface utilisateur. Prenez le temps de comprendre comment elle fonctionne (ce que font les boutons, ce que contient chaque champ, chaque étiquette, etc...). **Vous risquez de faire une bonne partie du travail pour rien si vous ne prenez pas le temps de consulter l'interface.**

Vous n'avez aucun test unitaire ni test d'interface à écrire, et aucun ne vous sera fourni.

3.2 Interface utilisateur

La maquette est en anglais, mais vous devrez traduire l'interface en français (obligatoire).

C'est ici que l'utilisateur saisit le texte à chiffrer ou à déchiffrer. Si le champ est vide, un « [placeholder](#) » est visible à la place. Seules les lettres de l'alphabet, incluant espaces et points, sont acceptées dans ce champ.

Les boutons suivants servent à chiffrer et déchiffrer ce que l'utilisateur a saisi.

La fine barre jaune ici est une [barre de progression](#) à durée indéterminée. Vous devrez [la rendre visible](#) durant chaque tâche de fond et [la cacher](#) lorsque le travail est terminé.

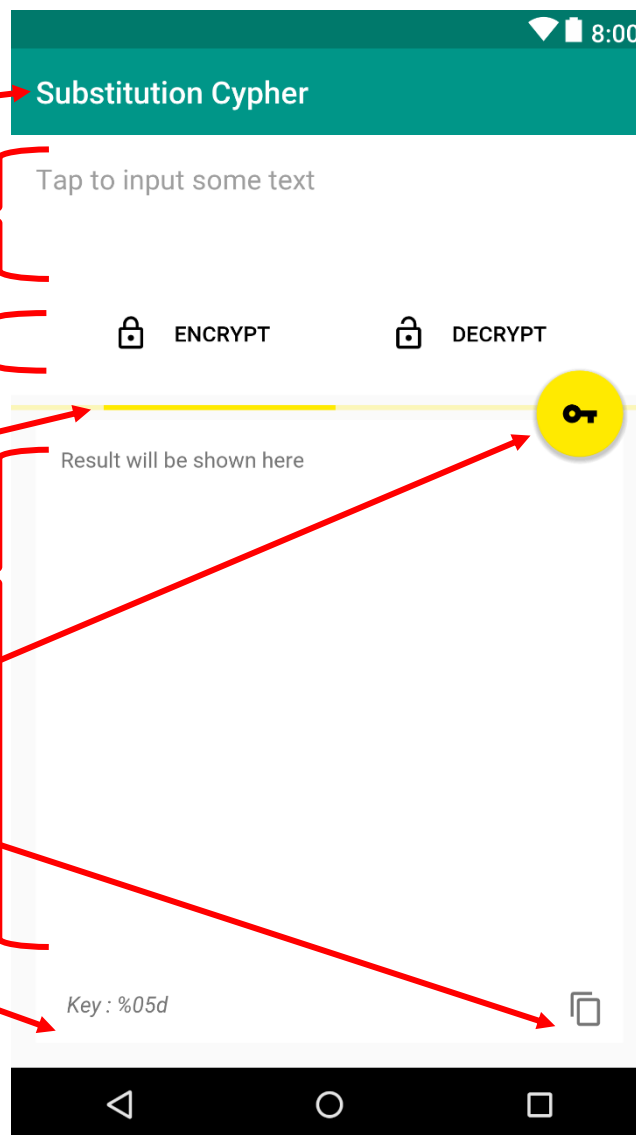
Le résultat du chiffrement ou du déchiffrement doit être inscrit ici. Il doit être possible de sélectionner le texte, mais pas de le modifier.

Ce gros [bouton rond](#) ouvre le dialogue de sélection de la clé.

Cette icône en bas à droite est en fait un bouton (plus précisément, un « [ImageButton](#) »). Il permet à l'utilisateur de copier le résultat dans le presse-papier. Un message de confirmation (avec « [Snackbar](#) ») doit être affiché pour confirmer que la copie a bien eu lieu.

Ce petit bout de texte affiche à l'utilisateur la clé qu'il utilise. Au lancement, une clé aléatoire est choisie. Attention à respecter le [format](#) demandé sur la maquette.

Cette interface devrait convenir au mode portrait et au mode paysage sans se déformer. Les données [ne doivent pas être perdues](#) après un changement d'orientation.



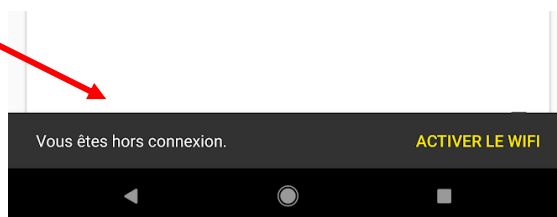
Le dialogue de sélection de la clé, lors de l'ouverture, devrait montrer la clé actuellement sélectionnée. Lors d'un changement d'orientation, le dialogue doit rester ouvert.

Le bouton « Ok » quitte le dialogue et débute le téléchargement de la nouvelle clé. Le bouton « Annuler » quitte le dialogue, mais ne modifie pas la clé.



Si une [erreur survient lors d'un appel réseau](#), un message doit apparaître en bas, [proposant à l'utilisateur des solutions](#) en fonction de l'erreur qui vient de survenir. Parmi les erreurs à gérer :

1. Pas de réseau : [Activer le Wifi](#) dans les [options](#).
2. [Erreur 500](#) du [serveur](#) : Faire patienter l'utilisateur (voir le fichier « strings.xml »).



3.3 Module de chiffrement

Votre application devra utiliser un [chiffrement par substitution](#) : chaque lettre est remplacée par une autre. Créez une couche « Modèle » pour répondre à ce besoin.

La clé de chiffrement est simple : il s'agit de deux tableaux dont vous vous servirez pour faire correspondre les caractères en fonction de leur position. Par exemple :

```
a b c d e f g h i j k l m n o p q r s t u v w x y z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z .  
x T H I a q Z O X V z v S M R L h m B l d e s K N n Q r p D o I g W b . G U E w C f P k A J t F c y u j Y
```

Les clés de chiffrement doivent être obtenues en effectuant un appel à [ce service web](#). Ici, le « id » est le numéro saisi dans le « KeyPickerDialog ».

```
https://m1t2.csfpwmjv.tk/api/key/{id}
```

Essayez-le dans votre navigateur quelques fois. Par exemple, pour le « id » « [01977](#) », vous obtiendrez :

```
{"id":1977,"outputCharacters":"xTHIaqZOXVzvSMRLhmB  
ldesKNnQrpDoigWb.GUEwCfPkAJtFcyujY","inputCharacters":"abcdefghijklmnopqrstuvwxyz  
ABCDEFGHIJKLMNOPQRSTUVWXYZ ."}]
```

Implémentez une tâche asynchrone qui effectue un appel à ce service web. Notez que ce dernier retourne du [« JSON »](#), alors vous devrez effectuer de la désérialisation avec [Jackson](#).

Pour d'autres informations sur ce service, vous pouvez aussi consulter [sa documentation](#).

4 Modalités de remise

Remettez sur LÉA à l'heure et à la date indiquée par votre professeur un fichier texte comportant l'adresse vers votre dépôt Git ou, si vous n'utilisez pas Git, votre projet complet sans les fichiers temporaires.

Incluez dans votre remise un fichier comprenant le nom et le matricule de tous les membres de votre équipe.

Une seule journée de retard est tolérée entraînant alors une pénalité de 15 % de la note. Au-delà de ce délai, le travail est refusé et la note « 0 » est attribuée.

5 Évaluation

Le travail sera évalué selon les critères suivants. Des points seront perdus en fonction de la gravité de la faute, de la qualité globale du travail et de l'effort mis par chaque membre de l'équipe dans ce travail.

Éléments
Fonctionnalités (incluant, mais sans s'y limiter) : <ul style="list-style-type: none">• Clé choisie aléatoirement au lancement de l'application.• Possibilité de choisir précisément une clé de chiffrement.• Utilisation d'un chiffrement par substitution.• Texte saisi crypté correctement avec la clé de chiffrement actuelle.• Texte saisi décrypté correctement avec la clé de chiffrement actuelle.• Possibilité de copier le résultat dans le presse-papier.• Caractères non supportés par le chiffrement refusés.• Messages d'erreurs se produisant lorsqu'une erreur de connexion se produit.
Interface utilisateur (incluant, mais sans s'y limiter) : <ul style="list-style-type: none">• Fichiers XML d'interface propres et bien structurés.• Visuel de l'interface utilisateur propre et stable.• Changement d'orientation et d'application supporté, sans perte de données.• Présence de messages d'erreurs clairs, avec solution, lorsque nécessaires.
Base de données (incluant, mais sans s'y limiter) : <ul style="list-style-type: none">• Obtention des données de l'application à partir du serveur fourni.• Gestion des erreurs de réseau et d'obtention des données.
Qualité générale de l'application mobile (incluant, mais sans s'y limiter) : <ul style="list-style-type: none">• Traitements lourds toujours effectués en tâche de fond.• Rétroaction visuelle au sujet des traitements en tâche de fond.
Qualité générale du code (incluant, mais sans s'y limiter) : <ul style="list-style-type: none">• Logique bien pensée, juste, et « non patché ».• Découpage adéquat du code en classes, méthodes et packages.• Séparation raisonnable des responsabilités entre les classes.• Respects des standards du langage de programmation.• Nommage clair et sans ambiguïté des éléments.• Utilisation de constantes, lorsque nécessaires.• Commentaires compensant uniquement le manque d'expressivité du code.• Respect des bonnes pratiques de programmation générales.• Code propre, sans résidus et facilement lisible.• Aucune erreur ni avertissement à la compilation.
Qualité générale du travail (incluant, mais sans s'y limiter) : <ul style="list-style-type: none">• Configuration Gradle fonctionnelle.• Respect des consignes de remise.• Aucun fichier inutile remis avec le projet.

Notez que la qualité écrite de la langue française fait partie intégrante de l'évaluation de ce travail. Étant donné que l'accès aux ressources de la langue française est permis, chaque faute de français retirera donc 0,5 % à la note finale jusqu'à concurrence de 20 % retirés. Sont corrigées les fautes d'orthographe, les fautes de grammaire, les fautes syntaxiques et les fautes de ponctuation.