Instructions:

1. Your app must meet the following minimum requirements and specifications described in

the tasks below. Deviating from these will result in a reduced grade.

2. Your app must validate its input without crashing.

3. This is an individual assignment, and no two students should have the same or similar code

for their XML and Java files. Feel free to discuss your approach with others, however, at a

high-level without sharing the code. In case you need a clarification, please feel free to

contact me.

4. You must use GitHub Classroom for this mini app. The invitation link is

https://classroom.github.com/a/s\_DxyHQi.

5. Make sure that your repository is private, and your repository MUST have at least 10 commits

with your GitHub username. Try to commit your work whenever you complete a method or

a significant part of the logic of your application. This is helpful especially in cases where you

would like to revert back to previous working code that you somehow modified.

Application Details and Specifications

For this Android Mini App, you are going to build a simple CYSEC212 – Donate app. On opening

the app, users will be able to log into this app and add donations (via check or cash) until a

target is meant. Since we haven’t covered data storage, the app will use hard coded credentials,

with a maximum of 3 authentication trial limits, and pre-populated list of donation records in

the report.

Your app will (most likely) have three .java files i.e.,

• LoginActivity.java – an Activity that lets the user log into the app

• DonateActivity.java – an Activity that lets the user donate by check or cash

• ReportActivity.java – an Activity that displays a static report of the donations made

You will also have the associated layout .xml files that go with these java files.

Please follow the tasks below to implement your app.

TASK 01: Design Activity Layouts

1. Design three activities in your application namely LoginActivity, DonateActivity and

ReportActivity as shown in the figure below.

a. Feel free to use any Android layout and views as long as your app is similar to the

screenshots shown in the figure below. The images used are provided in the project

template.

b. Your UI elements must be properly aligned and formatted

TASK 02: Implement App Logic

Implement the application logic according to the following specifications:

LoginActivity

• When the app starts up, the login screen (LoginActivity) should be the first one displayed.

• The user should be able to enter his credentials (i.e., username and password).

• The credentials should be validated against hard-coded username and password in the app.

The user has a maximum of three login attempts, otherwise, the application will close. The

number of Login Attempts Left should be displayed.

• The user can also click cancel thereby closing the application.

手順

1. アプリは、以下のタスクに記載されている最低要件と仕様を満たしている必要があります。

を満たしている必要があります。これらを逸脱した場合、減点の対象となります。

2. 2. アプリはクラッシュすることなく入力を検証する必要があります。

3. 3. これは個人的な課題であり、2人の学生が同じまたは類似のコードをXMLファイルとJavaファイルに使用することはできません。

3.これは個人的な課題であり、2人の学生がXMLとJavaのファイルに同じまたは類似のコードを持つべきではありません。ただし、コードを共有せず、高レベルで他の人とアプローチについて自由に議論してください。

ただし、コードを共有することなく、高レベルの議論をすることは自由です。不明な点がある場合は、お気軽に

私に連絡してください。

4. 4. このミニアプリでは、GitHub Classroomを使用する必要があります。招待リンクは

https://classroom.github.com/a/s\_DxyHQi。

5. 5. あなたのリポジトリが非公開であることを確認し、あなたのリポジトリには、あなたのGitHubユーザー名で少なくとも10件のコミットがなければなりません。

あなたのGitHubユーザー名で最低10回のコミットが必要です。メソッドやアプリケーションのロジックの重要な部分を完成させたときは、必ずコミットするようにしましょう。

メソッドやアプリケーションのロジックの重要な部分を完成させたときには、必ずコミットするようにしましょう。これは、特に次のような場合に役立ちます。

これは、特に、何らかの方法で変更したコードを以前のものに戻したい場合に便利です。

アプリケーションの詳細と仕様

この Android ミニアプリケーションでは、シンプルな CYSEC212 - Donate アプリを作成する予定です。アプリを開くと

このアプリを開くと、ユーザーはこのアプリにログインし、目標が達成されるまで（小切手または現金で）寄付を追加することができるようになります。

を追加することができます。データストレージをカバーしていないので、アプリはハードコードされたクレデンシャルを使用します。

認証の試行回数は最大3回までとし、寄付の記録はあらかじめレポートに入力されるようにします。

を使用します。

アプリには3つの.javaファイル（例）があります。

- LoginActivity.java - ユーザーがアプリにログインするためのアクティビティです。

- DonateActivity.java - ユーザーが小切手または現金で寄付をするためのアクティビティです。

- ReportActivity.java - 寄付を行った静的なレポートを表示するActivity

また、これらのJavaファイルに付随するレイアウト.xmlファイルも用意します。

以下のタスクに従って、アプリを実装してください。

TASK 01: アクティビティレイアウトの設計

1. アプリケーションの3つのアクティビティ、LoginActivity、DonateActivity、ReportActivity を設計してください。

ReportActivity を設計してください。

a. a. 下図のようなアプリであれば、Androidのレイアウトやビューを自由に使うことができます。

a. 下図に示すスクリーンショットのようなアプリであれば、Androidのレイアウトやビューは自由です。使用する画像は、プロジェクト

テンプレートで提供されています。

b. b. UI要素は適切に配置され、フォーマットされている必要があります。

TASK 02: アプリケーションロジックの実装

以下の仕様にしたがって、アプリケーションロジックを実装してください。

LoginActivity

- アプリの起動時に、最初に表示されるのがログイン画面（LoginActivity）です。

- ユーザーは、ユーザー名とパスワードを入力する必要があります。

- この認証情報は、アプリにハードコードされたユーザー名とパスワードと照合される必要があります。

ユーザーは最大3回までログインを試行することができ、それ以降はアプリケーションが終了します。ログイン試行回数

ログイン試行回数が表示される必要があります。

- ユーザーは、「キャンセル」をクリックして、アプリケーションを終了することもできます。

www.DeepL.com/Translator（無料版）で翻訳しました。

DonateActivity

• This activity should allow the user to add a Donation. The amount of the Donation could be

entered using NumberPicker or EditText above the Donate button.

a. The NumberPicker should have a minimum of 0 and maximum of 1,000.

b. By default, the amount in the NumberPicker should be used when adding the

donation. However, if the amount in the NumberPicker is equal to 0, then Amount

EditText should be used.

c. On addition of a donation, the Total So Far amount should be incremented by the

amount donated (even beyond the target) while the ProgressBar should be

incremented until a target 10,000 is reached.

d. Whenever a donation is made, a message should be displayed to the user thanking

him for the donation. Amount of the donation should be included in the thank you

message to the user.

e. When the target of 10,000 is achieved, a message of “Congrats!! Target

Exceeded…” should be displayed to the user.

• When the user selects the Check option, the Check details should be displayed and activated

for the user to take a picture of the check. Otherwise, the Check details should be invisible to

the user.

• The app needs to display the picture of the check without necessarily saving it.

• Extra credit will be given for enabling saving of the taken photo of the check in your app.

HINT: To do this, you will require the use of permissions.

ReportActivity

• This activity must be implemented with the help of a RecyclerView. For more details,

please refer to the Android Developer documentation at

https://developer.android.com/develop/ui/views/layout/recyclerview

• A hard-coded list of donations, in form of Payment Method – Amount (e.g., Check – 100,

Direct – 5000, etc.), should be displayed in this activity.

• On clicking an item in the list, the item details should be displayed

TASK 03: Create the Options Menu

Add an Options menu that appears on the top right corner of DonateActivity and

ReportActivity with two options:

a) Donate – starts the DonateActivity

b) Report – starts the ReportActivity

HINT: For details on how to create an Options menu, please refer to the Android Developer

documentation at <https://developer.android.com/guide/topics/ui/menus>

寄付アクティビティ（DonateActivity

- この活動は、ユーザーが寄付を追加することを可能にする必要があります。寄付の金額は、寄付ボタンの上にあるNumberPickerまたはEditTextを使用して入力することができます。

Donateボタンの上のNumberPickerまたはEditTextを使用して入力されるかもしれません。

a. ナンバーピッカーは、最小値0、最大値1,000に設定する必要があります。

b. b. デフォルトでは、NumberPickerの金額は、寄付を追加するときに使用されます。

b. デフォルトでは、寄付を追加するときに NumberPicker の金額が使用されます。ただし、NumberPickerの金額が0である場合は、Amount

EditText が使用されます。

c. c. 寄付を追加するとき、Total So Farの金額は、寄付された金額によって増加されるべきです。

c. 寄付が追加されると、Total So Farの金額は寄付された金額だけ(目標を超えても)増加し、ProgressBarは目標の10,000円まで増加する。

目標値である10,000に達するまで、ProgressBarはインクリメントされる。

d. 寄付がなされるたびに、ユーザーに対して寄付を感謝するメッセージが表示されるべきです。

d. 寄付が行われるたびに、寄付に対する感謝のメッセージを表示する。寄付の金額は、ユーザーへのお礼のメッセージに含まれるべきです。

メッセージに含める。

e. e. 10,000の目標が達成されたとき、"Congrats!!!"のメッセージを表示する。目標

Exceeded... "のメッセージをユーザーに表示する。

- ユーザーがチェックオプションを選択すると、チェックの詳細が表示され、ユーザーが写真を撮ることができるようにアクティブにする必要があります。

を表示し、ユーザーがチェックの写真を撮れるようにします。そうでない場合は、小切手の詳細はユーザーには表示されません。

を表示する必要があります。

- アプリは、小切手の写真を必ずしも保存せずに表示する必要があります。

- 撮影した小切手の写真をアプリに保存できるようにすると、さらに評価されます。

HINT: これを行うには、パーミッションを使用する必要があります。

レポートアクティビティ

- このアクティビティは、RecyclerViewの助けを借りて実装する必要があります。詳しくは、Android Developerドキュメントをご覧ください。

にあるAndroid Developerのドキュメントを参照してください。

https://developer.android.com/develop/ui/views/layout/recyclerview

- 支払い方法 - 金額の形式でハードコードされた寄付のリスト（例：小切手 - 100,

このアクティビティには、支払方法 - 金額（例：小切手 - 100、口座振替 - 5000など）の形式で、ハードコードされた寄付金リストが表示される必要があります。

- リスト内の項目をクリックすると、項目の詳細が表示されるはずです。

TASK 03: オプションメニューの作成

DonateActivityとReportActivityの右上に表示されるオプション・メニューを追加してください。

ReportActivityの右上に表示される、2つのオプションメニューを追加します。

a) Donate - DonateActivityを開始します。

b) Report - ReportActivityを開始します。

HINT: オプションメニューの作成方法の詳細については、Android Developerのドキュメントを参照してください。

のドキュメント（https://developer.android.com/guide/topics/ui/menus）を参照してください。

www.DeepL.com/Translator（無料版）で翻訳しました。