実技課題

実技課題

「多機能通信端末プログラミング」

|  |
| --- |
| １　作業時間  　　 ４１０分  ２　配付資料  ・問題用紙  ・課題で使用するデータ  ３　課題作成、提出方法  ・はじめに課題1を作成し、提出  ・課題2以降はプログラムを作成し、指導員の指定する場所に保存  ・課題3、4、5、6については、任意の順に課題を作成して良い  ・課題4、5は、それぞれ2種類の選択候補から選んで作成  ・課題が終了した時点で、指導員の確認を受けること  ※課題の中にある「画面例のイメージ」は、参考例です。各課題の要件を満たしていれば良いです。 |

課題１　UI設計

課題２～６の内容を確認してください。また、課題４および５について、それぞれ選択候補の１または２を選択してください。

作成するアプリケーション全体の画面遷移図、および各アクティビティの画面設計図を描き、提出してください。

課題２　ベース用Activitiy

課題３以降で作成するアプリケーションの、ベースとなるActivityを作成してください。ただし、下の要件を満たしてください。

【要件】

・アプリケーション起動時のActivity（MainActivityとします）には、4つのボタンを配置し、各ボタンに「背景色設定」、「センサー」、「サービス」、「ネットワーク」と表示させる。

参考

・「背景色設定」ボタンをタップすると、課題３が実現する。

・「センサー」ボタンをタップすると、課題４が実現する。

・「サービス」ボタンをタップすると、課題５が実現する。

・「ネットワーク」ボタンをタップすると、課題６が実現する。

課題３　背景色の設定

MainActivity（作成済み）の背景色を設定するアプリケーションを作成してください。ただし、下の要件を満たしてください。

【要件】（参照：図3-1）

・アプリケーションを起動すると、MainActivity（作成済み）が表示される。

・「背景色設定」ボタンをタップすると、別のActivity（SetColorActivity）が表示される。

・SetColorActivityには、白、赤、青、緑、黄の5色を選択できる仕組みと、「決定」ボタンを配置する。

・選択した色は、Intentに格納する。

・SetColorActivityの「決定」ボタンをタップすると、Intentに格納した色の背景色に変更されたMainActivityが表示される。



図3-1　状態遷移

課題４　センサー　【選択１】

水準器を実現するアプリケーションを作成してください。ただし、下の要件を満たしてください。

【要件】（参照：図4-1）

・アプリケーションを起動すると、MainActivity（作成済み）が表示される。

・「センサー」ボタンをタップすると、別のActivity（LevelActivity）が表示される。

・LevelActivityには、十字のラインを引き、水平状態で中心に丸印が配置されるようにする。また、水平状態からの角度に応じて丸印が移動する。（参照：図4-2）

・LevelActivityで「戻る」ボタンをタップすると、MainActivityが表示される。



図4-1　状態遷移



図4-2　SensorActivityの画面例のイメージ

課題４　センサー　【選択２】

照度計を実現するアプリケーションを作成してください。ただし、下の要件を満たしてください。

【要件】（参照：図4-3）

・アプリケーションを起動すると、MainActivity（作成済み）が表示される。

・「センサー」ボタンをタップすると、別のActivity（IlluminometerActivity）が表示される。

・IlluminometerActivityには、照度（単位：lux（ルクス））の測定結果の数値と、判定結果の評価を表示する。（参照：図4-4）

ここで評価は、デフォルト「－」、70未満「×」、70以上150未満「△」、150以上300未満「○」、300以上「◎」とする。

・IlluminometerActivityで「戻る」ボタンをタップすると、MainActivityが表示される。



図4-3　状態遷移



図4-4　SensorActivityの画面例のイメージ

課題５　サービス　【選択１】

Serviceコンポーネントを使用し、タイマーを実現するアプリケーションを作成してください。ただし、下の要件を満たしてください。

【要件】（参照：図5-1）

・アプリケーションを起動すると、MainActivity（作成済み）が表示される。

・「サービス」ボタンをタップすると、別のActivity（TimerActivity）が表示される。

・TimerActivityは、図5-2を参考にレイアウト構成し、BroadcastReceiverで残り時間データを受け取った時は中央のテキストに表示する。

・「スタート」ボタンを押すと、Service（TimerService）を起動し、そこでカウントダウンを開始する。  
1秒経過毎に、BroadcastReceiverに残り時間データを送る。カウントが０になると、音を鳴らし、サービスを停止する。

・カウントダウン中に「ストップ」ボタンを押すと、Service（TimerService）を停止する。カウントダウンも停止する。

・TimerActivityで「戻る」ボタンをタップすると、MainActivityが表示される。



図5-1　状態遷移



図5-2　TimerActivityの画面例のイメージ

課題５　サービス　【選択２】

Serviceコンポーネントを使用し、開始5秒後にNotificationを表示するアプリケーションを作成してください。ただし、下の要件を満たしてください。

【要件】（参照：図5-3）

・アプリケーションを起動すると、MainActivity（作成済み）が表示される。

・「サービス」ボタンをタップすると、別のActivity（NotificationActivity）が表示される。

・NotificationActivityには、「開始」ボタンと「停止」ボタンを配置する。

・「開始」ボタンをタップするとServiceが起動し、5秒後にNotificationが表示される。  
（参照：図5-4、図5-5）

・「停止」ボタンをタップするとServiceが停止し、Notificationが消える。

・NotificationActivityで「戻る」ボタンをタップすると、MainActivityが表示される。



図5-3　状態遷移



図5-4　NotificationActivityの画面例のイメージ



図5-5　Notificationの例

課題６　ネットワーク

実習環境下で閲覧可能な適当なWebページにアクセスし、HTTPレスポンスで得られたHTMLをテキストのまま表示するアプリケーションを作成してください。ただし、下の要件を満たしてください。

【要件】（参照：図6-1）

・アプリケーションを起動すると、MainActivity（作成済み）が表示される。

・「ネットワーク」ボタンをタップすると、別のActivity（NetActivity）が表示される。

・NetActivityには、適当なWebページのHTML文字列をそのまま表示させる。（参照：図6-2）

ここで、Webページのアドレスは、プログラムのソースコードに埋め込んで良いものとする。

・NetActivityで「戻る」ボタンをタップすると、MainActivityが表示される。



図6-1　状態遷移



図6-2　NetActivityの画面例のイメージ