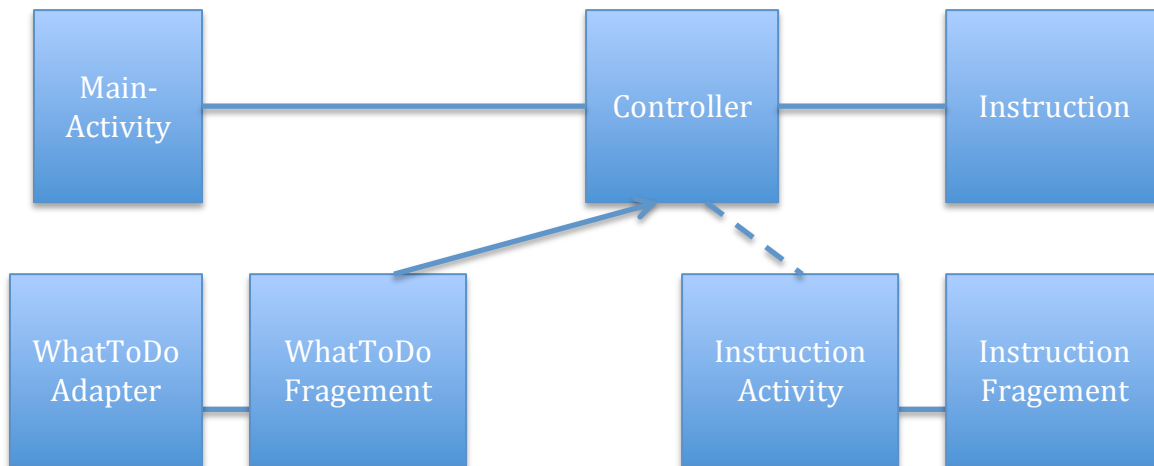


Laboration 4

Laboration 4a

Variera HowTo appen så att den fungerar med två Activity-klasser. MainActivity vilken visar lista med WhatToDo-strängar. Vid klick på listitem så ska InstructionActivity startas och visa motsvarande instruktion.

Mycket det du gjort i tidigare laborationer kan återanvändas. Designen beskrivs till viss del av figuren:



Att tänka på:

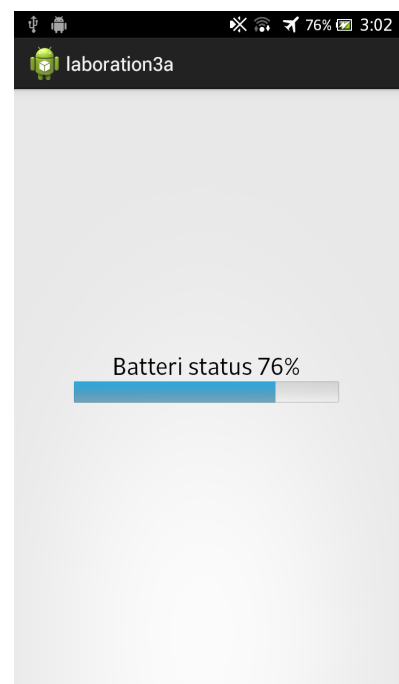
- Intentet som startar InstructionActivity ska innehålla info om vad som ska visas.
- InstructionActivity ska avläsa infon och placera infon i InstructionFragement. Det skulle kunna vara en Controller-klass även för InstructionActivity men det hoppar vi över.

Laboration 4b

Skapa en app med en ProgressBar och en TextView som lyssnar på batteri-status meddelanden från systemet. Din lösning skall minst innehålla en Activity-klass, en Fragment-klass, och en Controller-klass.

Funktion i app:

- Från början skall Progressbar vara vid 0 %
- Från början skall TextView innehålla "Batteri är på 0%"
- När batteristatus uppdateras av systemet skall Progressbar uppdateras med motsvarande värde.
- När batteristatus uppdateras av systemet skall TextView uppdateras med motsvarande värde

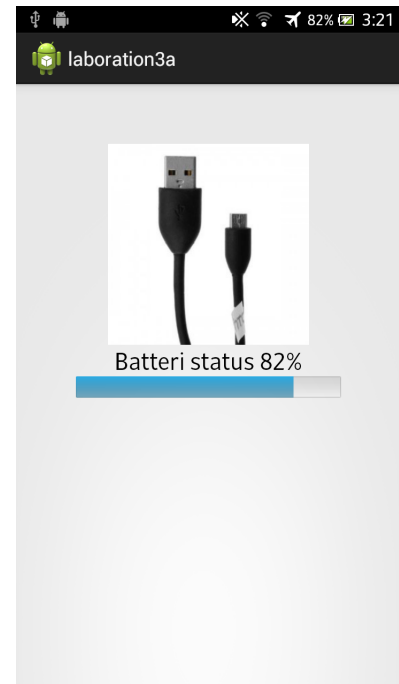


Laboration 4c

Fortsätt på laboration 4b. Lägg till en indikation på vilket laddningssätt som just nu används av telefonen.

Funktion i app:

- Appen skall visa en ikon som motsvarar laddningssättet, det kan vara tre olika typer av ikoner.
 - Ingen laddning pågår
 - Laddning via USB
 - Laddning via AC

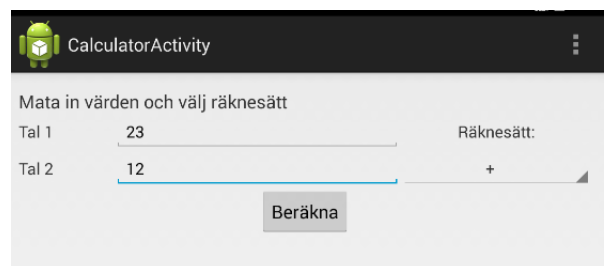


Laboration 4d

Din applikation ska bestå av två Activities, MainActivity och CalculatorActivity. CalculatorActivity ska startas via Intent då en beräkning behöver utföras. Beräkningarna ska vara enkla – bestå av två tal och någon av räkneoperationerna +, -, * eller /.

Funktion i app

- **Version 1.0**
Användaren klickar på en knapp i MainActivity varvid CalculatorActivity startas. CalculatorActivity visar ett enkelt gränssnitt som låter användaren mata in två tal och en räkneoperation. Då användaren klickar på Beräkna-knappen ska två värden returneras till MainActivity (om inmatning som i figuren till höger så):
 - * Resultatet av beräkningen (double): 35.0
 - * Sträng som visar beräkningen: 23+12=35.0



- **Version 1.1**
Användaren ska kunna skicka med två tal (double) och ett räknesätt till CalculatorActivity då den startas. Om alla tre värdena finns sker beräkning och resultatvärden returneras. Om något värde saknas ska appen fungera som i Version 1.0.

Det är fullt möjligt att CalculatorActivity avslutas utan att UI skapas och görs synligt. Det blir onCreate något liknande:

```
super.onCreate(...);  
CalculatorController controller = new CalculatorController(this);  
if(controller.validInput()) {
```

```

        controller.calculateAndReturn();
    } else {
        setContentView(...)
        :
    }
}

```

- **Version 1.2**

Kontrollera att värden är OK (går att göra till double) och att räknesätt är OK (+, -, *, eller /). Detta gäller främst värden som kommer via Intentet som startade activityn. Som version 1.1 men om det krävs inmatning av användaren så ska ev medskickade korrekta värden / räknesätt visas i Uiet. Så användaren slipper mata in dessa.

Laboration 4e

Lägg till intent-filter i manifestet för CalculatorActivity. Det kan se ut så här:

```

<activity android:name=".CalculatorActivity" >
    <intent-filter>
        <action android:name="se.mah.username.ACTION_CALC"/>
        <category android:name="android.intent.category.DEFAULT"/>
    </intent-filter>
</activity>

```

Ersätt username med ditt username.

Ändra i din lösning av 4d så att CalculatorActivity startas med ett implicit intent, ungefär:

```

Intent intent = new Intent("se.mah.username.ACTION_CALC");
:
activity.startActivityForResult(intent, MainController.REQUEST_CODE);

```

När du testat din lösning så skriver du en enkel app vilken använder CalculatorActivity via implicit Intent.