

LAMPIRAN

Lampiran 1 Source Code Tampilan Awal

```
1  <?xml version="1.0" encoding="utf-8"?>
6.  <RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
7.      android:layout_width="match_parent"
8.      android:layout_height="match_parent">
9.
10.     <ImageView
11.         android:layout_width="match_parent"
12.         android:layout_height="match_parent"
13.         android:scaleType="fitXY"
14.         android:adjustViewBounds="true"
15.         android:src="@drawable/gambar_splash"/>
16.
17.     <ImageView
18.         android:paddingTop="25dp"
19.         android:layout_width="200dp"
20.         android:layout_height="200dp"
21.         android:scaleType="fitCenter"
22.         android:layout_centerHorizontal="true"
23.         android:adjustViewBounds="true"
24.         android:src="@drawable/logofooter1"/>
25.     <LinearLayout
26.         android:layout_width="match_parent"
27.         android:layout_height="wrap_content"
28.         android:orientation="horizontal"
29.         android:gravity="center_horizontal"
30.         android:layout_alignParentBottom="true" >
31.         <TextView
32.             android:layout_width="wrap_content"
33.             android:layout_height="wrap_content"
34.             android:paddingBottom="30dp"
35.             android:textSize="24dp"
```



```
66.         android:layout_height="wrap_content"
67.         android:typeface="sans"
68.         android:textStyle="bold"
69.         android:gravity="center"
70.         android:textSize="35sp"
71.
       android:textAppearance="?android:attr/textAppearanceLarge"
72.
       android:text="@string/iftvmasukkandatasesuaikrs" />
73.
74.     <TextView
75.         android:id="@+id/textView1"
76.         android:layout_width="100dp"
77.         android:layout_height="wrap_content"
78.         android:layout_marginLeft="10dp"
79.         android:layout_marginTop="20dp"
80.         android:layout_below="@+id/textView0"
81.         android:typeface="sans"
82.         android:gravity="left"
83.         android:textSize="20sp"
84.         android:text="@string/iftvnm" />
85.
86.     <TextView
87.         android:id="@+id/textView4"
88.         android:layout_width="5dp"
89.         android:layout_height="wrap_content"
90.         android:layout_marginLeft="10dp"
91.         android:layout_marginTop="20dp"
92.         android:layout_below="@+id/textView0"
93.
       android:layout_toRightOf="@+id/textView1"
94.         android:typeface="sans"
95.         android:gravity="left"
96.         android:textSize="20sp"
97.         android:text="@string/iftvsep" />
98.
99.     <EditText
```

```
100.         android:id="@+id/ednim"
101.         android:layout_width="fill_parent"
102.         android:layout_height="wrap_content"
103.         android:hint="@string/ifahnnim"
104.         android:layout_marginTop="10dp"
105.         android:inputType="number"
106.         android:layout_below="@+id/textView0"
107.
        android:layout_toRightOf="@+id/textView4"
108.         android:padding="10dp"
109.
        android:textCursorDrawable="@drawable/color_cursor" />
110.
111.         <TextView
112.             android:id="@+id/textView2"
113.             android:layout_width="100dp"
114.             android:layout_height="wrap_content"
115.             android:layout_marginLeft="10dp"
116.             android:layout_marginTop="25dp"
117.             android:layout_below="@+id/textView1"
118.             android:typeface="sans"
119.             android:gravity="left"
120.             android:textSize="20sp"
121.             android:text="@string/iftvpass"/>
122.
123.         <TextView
124.             android:id="@+id/textView6"
125.             android:layout_width="5dp"
126.             android:layout_height="wrap_content"
127.             android:layout_marginLeft="10dp"
128.             android:layout_marginTop="20dp"
129.             android:layout_below="@+id/textView4"
130.
            android:layout_toRightOf="@+id/textView2"
131.             android:typeface="sans"
132.             android:gravity="left"
133.             android:textSize="20sp"
```

```
134.         android:text="@string/iftvsep" />
135.
136.         <EditText
137.             android:id="@+id/edpassword"
138.             android:layout_width="fill_parent"
139.             android:layout_height="wrap_content"
140.             android:layout_below="@+id/ednim"
141.             android:layout_marginTop="5dp"
142.
143.             android:layout_toRightOf="@+id/textView6"
144.             android:hint="@string/iftvpass"
145.             android:inputType="textPassword"
146.             android:padding="10dp"
147.
148.             android:textCursorDrawable="@drawable/color_cursor" />
149.
150.         <CheckBox
151.             android:id="@+id/cbtampilkanpass"
152.             android:layout_width="wrap_content"
153.             android:layout_height="wrap_content"
154.
155.             android:layout_alignLeft="@+id/edpassword"
156.             android:layout_below="@+id/edpassword"
157.             android:text="Tampilkan Password" />
158.
159.         <Button
160.             android:id="@+id/btnsimpan"
161.             android:layout_width="match_parent"
162.             android:layout_height="wrap_content"
163.             android:layout_alignParentEnd="true"
164.             android:layout_alignParentLeft="true"
165.             android:layout_alignParentRight="true"
166.             android:layout_alignParentStart="true"
167.             android:layout_marginTop="20dp"
168.             android:background="@drawable/btnbg"
169.
170.             android:layout_below="@+id/cbtampilkanpass"
```

```

167.         android:text="@string/btnlogin"
168.     />
169.
170.     <ProgressBar
171.         android:layout_marginTop="15sp"
172.         android:layout_width="wrap_content"
173.         android:layout_height="wrap_content"
174.         android:layout_below="@+id/btnsimpan"
175.         android:layout_centerHorizontal="true"
176.         android:id="@+id/pbbar" />
177.     </RelativeLayout>
178. </ScrollView>
179. </RelativeLayout>

```

Lampiran 3 Source Code Tampilan Validasi Token

```

1 <RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
180.     xmlns:tools="http://schemas.android.com/tools"
181.     android:layout_width="match_parent"
182.     android:layout_height="match_parent"
183.     android:orientation="vertical">
184.
185.     <ScrollView
186.         android:layout_width="match_parent"
187.         android:layout_height="match_parent">
188.
189.         <LinearLayout
190.             android:layout_width="match_parent"
191.             android:layout_height="wrap_content"
192.             android:orientation="vertical"
193.             android:layout_alignParentTop="true"
194.             android:padding="2dp">
195.
196.             <LinearLayout
197.                 android:layout_width="match_parent"

```

```
198.         android:layout_height="wrap_content"
199.         android:orientation="horizontal"
200.         android:layout_alignParentTop="true"
201.         android:padding="2dp">
202.         <TextView
203.             android:layout_width="wrap_content"
204.             android:layout_height="match_parent"
205.             android:text="ASSALAMU'ALAIKUM, "
206.             android:layout_marginTop="7dp"
207.             android:typeface="sans"
208.             android:textSize="15sp"
209.             android:gravity="left"/>
210.
211.         <TextView
212.             android:layout_width="wrap_content"
213.             android:layout_height="match_parent"
214.             android:text=""
215.             android:layout_marginTop="7dp"
216.             android:typeface="sans"
217.             android:textSize="15sp"
218.             android:gravity="left"
219.             android:textAllCaps="true"
220.             android:id="@+id/username" />
221.
222.     </LinearLayout>
223.
224.     <View
225.         android:layout_width="match_parent"
226.         android:layout_height="2dp"
227.         android:background="#c0c0c0"/>
228.
229.     <TextView
230.         android:layout_width="match_parent"
231.         android:layout_height="wrap_content"
232.         android:text="Masukkan Token"
233.         android:layout_marginTop="7dp"
234.         android:typeface="sans"
```

```
235.         android:textSize="35sp"
236.         android:gravity="center"/>
237.
238.     <EditText
239.         android:layout_width="match_parent"
240.         android:layout_height="wrap_content"
241.         android:layout_marginTop="10dp"
242.         android:textStyle="bold"
243.         android:background="#AAB7B8"
244.         android:padding="10dp"
245.         android:hint="Token"
246.         android:gravity="center"
247.         android:textSize="20sp"
248.
249.         android:textCursorDrawable="@drawable/color_cursor"
250.         android:id="@+id/edtoken" />
251.
252.     <LinearLayout
253.         android:layout_width="match_parent"
254.         android:layout_height="wrap_content"
255.         android:weightSum="3"
256.         android:layout_marginTop="10dp"
257.         android:gravity="center"
258.         android:orientation="horizontal">
259.
260.         <Button
261.             android:layout_marginTop="10dp"
262.             android:layout_width="match_parent"
263.             android:layout_height="wrap_content"
264.             android:layout_weight="1"
265.             android:background="@drawable/btnbg"
266.
267.             android:textSize="20sp"
268.             android:layout_margin="2dp"
269.             android:padding="7dp"
270.             android:layout_gravity="center"
271.             android:id="@+id/btnadd"
```



```

271.             android:text="Submit" />
272.         </LinearLayout>
273.
274.         <ProgressBar
275.             android:layout_width="wrap_content"
276.             android:layout_height="wrap_content"
277.             android:layout_gravity="center"
278.             android:id="@+id/pbbar" />
279.     </LinearLayout>
280. </ScrollView>
281. </RelativeLayout>

```

Lampiran 4 *Source Code Tampilan Koneksi Error*

```

1  <RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
282.     xmlns:tools="http://schemas.android.com/tools"
283.     android:layout_width="fill_parent"
284.     android:layout_height="wrap_content"
285.     android:orientation="vertical"
286.     >
287.
288.     <LinearLayout
289.         android:layout_width="fill_parent"
290.         android:layout_height="wrap_content"
291.         android:orientation="vertical"
292.         android:layout_alignParentTop="true"
293.         android:padding="2dp">
294.
295.         <TextView
296.             android:layout_width="fill_parent"
297.             android:layout_height="wrap_content"
298.             android:layout_marginTop="7dp"
299.             android:typeface="sans"
300.             android:textSize="35sp"
301.

```

```
302.         android:text="Terjadi Kesalahan Dengan
           Jaringan"
303.         android:gravity="center"
304.         android:id="@+id/tvJudul" />
305.
306.
307.         <TextView
308.             android:layout_width="fill_parent"
309.             android:layout_height="wrap_content"
310.             android:text="Mohon Sambungkan Dengan
           Jaringan UMY !!!"
311.             android:layout_marginTop="15dp"
312.             android:typeface="sans"
313.             android:textSize="20sp"
314.
315.             android:id="@+id/tvpesan"
316.             android:visibility="visible"
317.             android:gravity="center"
318.         />
319.
320.         <LinearLayout
321.             android:layout_width="fill_parent"
322.             android:layout_height="wrap_content"
323.             android:layout_marginTop="25dp"
324.             android:orientation="horizontal"
325.             android:layout_alignParentTop="true"
326.             android:padding="2dp">
327.
328.             <Button
329.                 android:id="@+id/btnok"
330.                 android:layout_width="0dp"
331.                 android:layout_height="wrap_content"
332.                 android:layout_marginRight="5dp"
333.                 android:layout_marginTop="10dp"
334.                 android:layout_weight="1"
335.                 android:background="@drawable/btnbg"
336.                 android:padding="10dp"
```

```

337.             android:text="Coba Lagi"
338.             android:textSize="20sp"
339.             android:textStyle="bold" />
340.
341.         <Button
342.             android:id="@+id/btncancel"
343.             android:layout_width="0dp"
344.             android:layout_height="wrap_content"
345.             android:layout_marginTop="10dp"
346.
347.             android:layout_toRightOf="@+id/btnsubmit"
348.             android:layout_weight="1"
349.             android:background="@drawable/btnbg"
350.             android:padding="10dp"
351.             android:text="Batal"
352.             android:textSize="20sp"
353.             android:textStyle="bold" />
354.     </LinearLayout>
355.     <ProgressBar
356.         android:layout_width="wrap_content"
357.         android:layout_height="wrap_content"
358.         android:layout_gravity="center"
359.         android:id="@+id/pbbar" />
360. </LinearLayout>
</RelativeLayout>

```

Lampiran 5 Source Code Tampilan Detail Presensi

```

1 <LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
361.     xmlns:tools="http://schemas.android.com/tools"
362.     android:layout_width="fill_parent"
363.     android:layout_height="fill_parent"
364.     android:minHeight="400dp"
365.     android:minWidth="300dp"
366.     android:orientation="vertical" >

```

```
367.
368.     <ScrollView
369.         android:layout_width="match_parent"
370.         android:layout_height="match_parent"
371.         android:fillViewport="true" >
372.
373.     <LinearLayout
374.         android:layout_width="match_parent"
375.         android:layout_height="wrap_content"
376.         android:orientation="vertical"
377.         android:padding="2dp" >
378.
379.     <TextView
380.         android:id="@+id/tvNmMk"
381.         android:layout_width="match_parent"
382.         android:layout_height="wrap_content"
383.         android:gravity="center"
384.         android:text="Nama Mata Kuliah:"
385.         android:textSize="20sp"
386.         android:typeface="sans" />
387.
388.     <TextView
389.         android:id="@+id/tvNamaMK"
390.         android:layout_width="match_parent"
391.         android:layout_height="wrap_content"
392.         android:layout_below="@+id/tvNmMk"
393.         android:gravity="center"
394.         android:text="Nama Mata Kuliah1"
395.         android:textSize="20sp"
396.         android:textStyle="bold"
397.         android:typeface="sans" />
398.
399.     <TextView
400.         android:id="@+id/tvNmKl"
401.         android:layout_width="match_parent"
402.         android:layout_height="wrap_content"
403.         android:layout_below="@+id/tvNamaMK"
```

```
404.         android:layout_marginTop="10dp"
405.         android:gravity="center"
406.         android:text="Kelas:"
407.         android:textSize="20sp"
408.         android:typeface="sans" />
409.
410.     <TextView
411.         android:id="@+id/tvNamaKl"
412.         android:layout_width="match_parent"
413.         android:layout_height="wrap_content"
414.         android:layout_below="@+id/tvNmKl"
415.         android:gravity="center"
416.         android:text="Kelas1"
417.         android:textSize="20sp"
418.         android:textStyle="bold"
419.         android:typeface="sans" />
420.
421.     <TextView
422.         android:id="@+id/tvNmDs"
423.         android:layout_width="match_parent"
424.         android:layout_height="wrap_content"
425.         android:layout_below="@+id/tvNamaKl"
426.         android:layout_marginTop="10dp"
427.         android:gravity="center"
428.         android:text="Dosen Pengampu: "
429.         android:textSize="20sp"
430.         android:typeface="sans" />
431.
432.     <TextView
433.         android:id="@+id/tvNamaDs"
434.         android:layout_width="match_parent"
435.         android:layout_height="wrap_content"
436.         android:layout_below="@+id/tvNmDs"
437.         android:gravity="center"
438.         android:text="Dosen Pengampul"
439.         android:textSize="20sp"
440.         android:textStyle="bold"
```

```
441.         android:typeface="sans" />
442.
443.         <TextView
444.             android:id="@+id/tvNmSs"
445.             android:layout_width="match_parent"
446.             android:layout_height="wrap_content"
447.             android:layout_below="@+id/tvNamaDs"
448.             android:layout_marginTop="10dp"
449.             android:gravity="center"
450.             android:text="Sesi: "
451.             android:textSize="20sp"
452.             android:typeface="sans" />
453.
454.         <TextView
455.             android:id="@+id/tvNamaSs"
456.             android:layout_width="match_parent"
457.             android:layout_height="wrap_content"
458.             android:layout_below="@+id/tvNmSs"
459.             android:gravity="center"
460.             android:text="Sesil"
461.             android:textSize="20sp"
462.             android:textStyle="bold"
463.             android:typeface="sans" />
464.
465.         <TextView
466.             android:id="@+id/tvNmHrTgl"
467.             android:layout_width="match_parent"
468.             android:layout_height="wrap_content"
469.             android:layout_below="@+id/tvNamaSs"
470.             android:layout_marginTop="10dp"
471.             android:gravity="center"
472.             android:text="Hari / Tanggal"
473.             android:textSize="20sp"
474.             android:typeface="sans" />
475.
476.         <LinearLayout
477.             android:id="@+id/lin"
```

```
478.             android:layout_width="match_parent"
479.             android:layout_height="wrap_content"
480.
481.             android:layout_below="@+id/tvNmHrTgl"
482.             android:gravity="center"
483.             android:orientation="horizontal" >
484.             <TextView
485.                 android:id="@+id/tvNamaHr"
486.
487.                 android:layout_width="wrap_content"
488.                 android:layout_height="wrap_content"
489.                 android:layout_below="@+id/tvNmHrTgl"
490.                 android:gravity="center"
491.                 android:text="Hari"
492.                 android:textSize="20sp"
493.                 android:textStyle="bold"
494.                 android:typeface="sans" />
495.             <TextView
496.                 android:id="@+id/tvappos"
497.
498.                 android:layout_width="wrap_content"
499.                 android:layout_height="wrap_content"
500.                 android:layout_below="@+id/tvNmHrTgl"
501.                 android:gravity="center"
502.                 android:text=", "
503.                 android:textSize="20sp"
504.                 android:textStyle="bold"
505.                 android:typeface="sans" />
506.             <TextView
507.                 android:id="@+id/tvNamaTgl"
```

```
508.         android:layout_width="wrap_content"
509.
510.         android:layout_height="wrap_content"
511.
512.         android:layout_below="@+id/tvNmHrTgl"
513.
514.         android:layout_toRightOf="@+id/tvNamaHr"
515.
516.         android:gravity="center"
517.         android:text="Tanggal"
518.         android:textSize="20sp"
519.         android:textStyle="bold"
520.         android:typeface="sans" />
521.     </LinearLayout>
522.
523.     <TextView
524.         android:id="@+id/tvjm"
525.         android:layout_width="match_parent"
526.         android:layout_height="wrap_content"
527.         android:layout_below="@+id/lin"
528.         android:layout_marginTop="10dp"
529.         android:gravity="center"
530.         android:text="Mulai: "
531.         android:textSize="20sp"
532.         android:typeface="sans" />
533.
534.     <TextView
535.         android:id="@+id/tvJmMulai"
536.         android:layout_width="match_parent"
537.         android:layout_height="wrap_content"
538.         android:layout_below="@+id/tvjm"
539.         android:gravity="center"
540.         android:text="Jam"
541.         android:textSize="20sp"
542.         android:textStyle="bold"
543.         android:typeface="sans" />
```



```

541.          <TextView
542.              android:id="@+id/tvmakter"
543.              android:layout_width="match_parent"
544.              android:layout_height="wrap_content"
545.
546.              android:layout_below="@+id/tvJmMulai"
547.              android:layout_marginTop="10dp"
548.              android:gravity="center"
549.              android:text="Maksimal
550.              Keterlambatan: "
551.              android:textSize="20sp"
552.              android:typeface="sans" />
553.
554.          <TextView
555.              android:id="@+id/tvmt"
556.              android:layout_width="match_parent"
557.              android:layout_height="wrap_content"
558.              android:layout_below="@+id/tvmakter"
559.              android:gravity="center"
560.              android:text="Maksimal
561.              Keterlambatan1"
562.              android:textSize="20sp"
563.              android:textStyle="bold"
564.              android:typeface="sans" />
565.
566.          <LinearLayout
567.              android:layout_width="fill_parent"
568.              android:layout_height="fill_parent"
569.              android:layout_alignParentTop="true"
570.              android:layout_marginTop="25dp"
571.              android:orientation="horizontal"
572.              android:padding="2dp" >
573.
574.          <Button
575.              android:id="@+id/btnok"
576.              android:layout_width="0dp"

```

```

574.         android:layout_height="wrap_content"
575.                 android:layout_marginRight="5dp"
576.                 android:layout_marginTop="10dp"
577.                 android:layout_weight="1"
578.
579.         android:background="@drawable/btnbg"
580.                 android:padding="10dp"
581.                 android:text="Presensi"
582.                 android:textSize="20sp"
583.                 android:textStyle="bold" />
584.         <Button
585.                 android:id="@+id/btncancel"
586.                 android:layout_width="0dp"
587.
588.                 android:layout_height="wrap_content"
589.                 android:layout_marginTop="10dp"
590.                 android:layout_toRightOf="@+id/btnsubmit"
591.                 android:layout_weight="1"
592.                 android:background="@drawable/btnbg"
593.                 android:padding="10dp"
594.                 android:text="Batal"
595.                 android:textSize="20sp"
596.                 android:textStyle="bold" />
597.         </LinearLayout>
598.     </LinearLayout>
599. </ScrollView>

```

Lampiran 6 Source Code String

```

1  <?xml version="1.0" encoding="utf-8"?>
600.     <resources>

```

```

601.
602.         <string name="app_name">Presensi Online</string>
603.         <string name="help">Bantuan</string>
604.         <string name="action_settings">tes</string>
605.         <string name="label_login"> Presensi Mahasiswa
        Online</string>
606.         <string name="iftvsep"> : </string>
607.         <string
        name="iftvmasukdatasesuaikrs">Masukkan Data Sesuai
        Dengan KRS</string>
608.         <string name="iftvdatapresensi">Detail Presensi
        </string>
609.         <string name="iftvkode">Kode Token</string>
610.         <string name="iftvnim">NIM </string>
611.         <string name="iftvnama">Nama </string>
612.         <string name="iftvpass">Password </string>
613.         <string name="ifhnnim">Masukkan NIM </string>
614.         <string name="ifhnnama">Masukkan Nama</string>
615.         <string name="ifhnpass">Masukkan
        Password</string>
616.         <string name="btnlogin">Login</string>
617.         <string name="btnsbm">Submit</string>
618.
619.         </resources>

```

Lampiran 7 Source Code DbController java

```

1  package com.example.presensionline;
620.
621.     import java.util.ArrayList;
622.     import java.util.HashMap;
623.     import android.content.ContentValues;
624.     import android.content.Context;
625.     import android.database.Cursor;
626.     import android.database.sqlite.SQLiteDatabase;
627.     import android.database.sqlite.SQLiteOpenHelper;
628.

```

```
629.
630.     public class DBController extends SQLiteOpenHelper {
631.
632.         public DBController(Context
        applicationcontext) {
633.             super(applicationcontext, "datapemilik.db",
        null, 1);
634.         }
635.
636.         @Override
637.         public void onCreate(SQLiteDatabase database)
        {
638.             String query;
639.             query = "CREATE TABLE mahasiswa (id
        INTEGER PRIMARY KEY, nim INTEGER,nama TEXT,password
        TEXT)";
640.             database.execSQL(query);
641.         }
642.         @Override
643.         public void onUpgrade(SQLiteDatabase database,
        int version_old, int current_version) {
644.             String query;
645.             query = "DROP TABLE IF EXISTS sahabat";
646.             database.execSQL(query);
647.             onCreate(database);
648.         }
649.
650.         public void insertMahasiswa(HashMap<String,
        String> queryValues) {
651.             SQLiteDatabase database =
        this.getWritableDatabase();
652.             ContentValues values = new
        ContentValues();
653.             values.put("nim",
        queryValues.get("nim"));
654.             values.put("nama",
        queryValues.get("nama"));
```

```
655.             values.put("password",
        queryValues.get("password"));
656.             database.insert("mahasiswa", null,
        values);
657.             database.close();
658.         }
659.
660.         public ArrayList<HashMap<String, String>>
        getDataMahasiswa(int id) {
661.             ArrayList<HashMap<String, String>>
        mahasiswaList;
662.             mahasiswaList = new
        ArrayList<HashMap<String, String>>();
663.             String selectQuery = "SELECT  nim, nama
        FROM mahasiswa where id='"+'''+ id +''''";
664.             SQLiteDatabase database =
        this.getWritableDatabase();
665.             Cursor cursor =
        database.rawQuery(selectQuery, null);
666.             if (cursor.moveToFirst()) {
667.                 do {
668.                     HashMap<String, String> map = new
        HashMap<String, String>();
669.                     map.put("nim",
        cursor.getString(0));
670.                     map.put("nama",
        cursor.getString(1));
671.                     mahasiswaList.add(map);
672.                 } while (cursor.moveToNext());
673.             }
674.             database.close();
675.             return mahasiswaList;
676.         }
677.
678.         public ArrayList<HashMap<String, String>>
        getLoginMahasiswa(int id,String Nim, String pass) {
```

```

679.                ArrayList<HashMap<String, String>>
                    mahasiswaList;
680.                mahasiswaList = new
                    ArrayList<HashMap<String, String>>();
681.                String selectQuery = "SELECT  nim, nama,
                    password FROM mahasiswa where id='"+'''+ id +' ' and nim =
                    '"+'''+ Nim +' ' and password='"+'''+ pass +' '";
682.                SQLiteDatabase database =
                    this.getWritableDatabase();
683.                Cursor cursor =
                    database.rawQuery(selectQuery, null);
684.                if (cursor.moveToFirst()) {
685.                    do {
686.                        HashMap<String, String> map = new
                            HashMap<String, String>();
687.                        map.put("nim",
                            cursor.getString(0));
688.                        map.put("nama",
                            cursor.getString(1));
689.                        mahasiswaList.add(map);
690.                    } while (cursor.moveToNext());
691.                }
692.                database.close();
693.                return mahasiswaList;
694.            }
695.        }

```

Lampiran 8 Source Code Tambah Data Pemilik *Class java*

```

1  package com.example.presensionline;
2
3  import java.sql.Connection;
4  import java.sql.ResultSet;
5  import java.sql.Statement;
6  import java.util.HashMap;
7
8
9  import android.app.Activity;
10 import android.app.AlertDialog;
11 import android.app.Dialog;
12 import android.content.Context;

```

```

13 import android.content.DialogInterface;
14 import android.content.Intent;
15 import android.os.AsyncTask;
16 import android.os.Bundle;
17 import android.text.method.HideReturnsTransformationMethod;
18 import android.text.method.PasswordTransformationMethod;
19 import android.view.View;
20 import android.view.Window;
21 import android.view.View.OnClickListener;
22 import android.widget.Button;
23 import android.widget.CheckBox;
24 import android.widget.CompoundButton;
25 import android.widget.EditText;
26 import android.widget.ProgressBar;
27 import android.widget.Toast;
28 import android.widget.CompoundButton.OnCheckedChangeListener;
29
30 public class TambahDataPemilikClass extends Activity {
31
32     DBController controller = new DBController(this);
33     ConnectionClass connectionClass;
34
35     final Context context = this;
36     EditText editNim, editPassword;
37     Button buttonLogin;
38     ProgressBar pbar;
39     CheckBox cbshow;
40
41     String z = "";
42     String NamaMhs;
43     Boolean isSuccess = false;
44
45     protected void onCreate(Bundle savedInstanceState) {
46         super.onCreate(savedInstanceState);
47         setContentView(R.layout.registrasi_layout);
48
49         pbar = (ProgressBar) findViewById(R.id.pbar);
50         pbar.setVisibility(View.GONE);
51         connectionClass = new ConnectionClass();
52         editNim = (EditText) findViewById(R.id.ednim);
53         editPassword =
54         (EditText) findViewById(R.id.edpassword);
55         buttonLogin = (Button) findViewById(R.id.btnsimpan);
56         cbshow = (CheckBox)
57         findViewById(R.id.cbtampilkanpass);
58
59         cbshow.setOnCheckedChangeListener(new
60         OnCheckedChangeListener() {
61             @Override
62             public void onCheckedChanged(CompoundButton
63             buttonView, boolean isChecked) {
64                 // TODO Auto-generated method stub
65                 if (!isChecked) {
66                     // tampilkan password

```

```
63         editPassword.setTransformationMethod(PasswordTransformationMethod.getInstance());
64     } else {
65         // sembunyikan password
66         editPassword.setTransformationMethod(HideReturnsTransformationMethod.getInstance());
67     }
68     }
69     });
70
71     buttonLogin.setOnClickListener(new
View.OnClickListener() {
72
73         public void onClick(View v) {
74
75             if
(editNim.getText().toString().trim().equals("") ||
editPassword.getText().toString().trim().equals(""))
76         {
77             android.app.AlertDialog.Builder alert =
new AlertDialog.Builder(TambahDataPemilikClass.this);
78             alert.setTitle("Login Gagal");
79             alert.setMessage("NIM dan Password Tidak
Boleh Kosong");
80
81             alert.setPositiveButton("Ulangi", new
DialogInterface.OnClickListener() {
82                 public void
onClick(DialogInterface dialog, int id) {
83                     dialog.cancel();
84                 }
85             });
86             alert.show();
87         }
88         else if
(editNim.getText().toString().trim().equals(""))
89         {
90             android.app.AlertDialog.Builder alert =
new AlertDialog.Builder(TambahDataPemilikClass.this);
91             alert.setTitle("Login Gagal");
92             alert.setMessage("NIM Tidak Boleh
Kosong");
93
94             alert.setPositiveButton("Ulangi", new
DialogInterface.OnClickListener() {
95                 public void
onClick(DialogInterface dialog, int id) {
96                     dialog.cancel();
97                 }
98             });
99             alert.show();
100         }
101     }
```



```

102         else if
(editPassword.getText().toString().trim().equals(""))
103         {
104             android.app.AlertDialog.Builder alert =
new AlertDialog.Builder(TambahDataPemilikClass.this);
105             alert.setTitle("Login Gagal");
106             alert.setMessage("Password Tidak Boleh
Kosong");
107
108             alert.setPositiveButton("Ulangi", new
DialogInterface.OnClickListener() {
109                 public void
onClick(DialogInterface dialog, int id) {
110                     dialog.cancel();
111                 }
112             });
113             alert.show();
114         }
115         else if (editNim.getText().length() != 11)
116         {
117             android.app.AlertDialog.Builder alert =
new AlertDialog.Builder(TambahDataPemilikClass.this);
118             alert.setTitle("Login Gagal");
119             alert.setMessage("Format NIM
Salah"+"\\n"+"Mohon Masukkan NIM dengan Benar");
120
121             alert.setPositiveButton("Ulangi", new
DialogInterface.OnClickListener() {
122                 public void
onClick(DialogInterface dialog, int id) {
123                     dialog.cancel();
124                 }
125             });
126             alert.show();
127         }
128         else
129         {
130             {
131                 DoLogin doLogin = new DoLogin();
132                 doLogin.execute("");
133             }
134         }
135     });
136 }
137 //button exit
138 public void onBackPressed() {
139     AlertDialog.Builder builder = new
AlertDialog.Builder(this,
AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
140     builder.setMessage("Apakah Anda Ingin Keluar?")
141         .setCancelable(false)
142         .setPositiveButton("Iya", new
DialogInterface.OnClickListener() {
143             public void onClick(DialogInterface
dialog, int id) {
144                 TambahDataPemilikClass.this.finish();

```

```

145         }
146     })
147     .setNegativeButton("Tidak", new
DialogInterface.OnClickListener() {
148         public void onClick(DialogInterface
dialog, int id) {
149             dialog.cancel();
150         }
151     });
152     AlertDialog alert = builder.create();
153     alert.show();
154 }
155
156 public class DoLogin extends
AsyncTask<String,String,String>
157 {
158     String z = "";
159     Boolean isSuccess = false;
160
161
162     String enim = editNim.getText().toString();
163     String epassword = editPassword.getText().toString();
164
165
166     @Override
167     protected void onPreExecute() {
168         pbbbar.setVisibility(View.VISIBLE);
169     }
170
171     @Override
172     protected void onPostExecute(String r) {
173         pbbbar.setVisibility(View.GONE);
174
175         Toast.makeText(TambahDataPemilikClass.this,r,Toast.LENGTH_LONG).
show();
176
177         if(isSuccess) {
178             try {
179                 Connection con = connectionClass.CONN();
180                 if (con == null) {
181                     final Dialog dialog = new
Dialog(context);
182
183                     dialog.requestWindowFeature(Window.FEATURE_LEFT_ICON);
184
185                     dialog.setFeatureDrawableResource(Window.FEATURE_LEFT_ICON
, R.drawable.logoumy);
186
187                     dialog.setContentView(R.layout.errorkoneksi_layout);
188
189                     dialog.setCancelable(false);
190
191                     dialog.setCanceledOnTouchOutside(false);
192
193                     Button ok = (Button)
dialog.findViewById(R.id.btnok);

```

```

189             Button cancel = (Button)
dialog.findViewById(R.id.btncancel);
190
191             ok.setOnClickListener(new
OnClickListener() {
192
193                 public void onClick(View
arg0) {
194
195                     TambahDataPemilikClass.this.recreate();
196                 }
197             });
198             cancel.setOnClickListener(new
OnClickListener() {
199
200                 public void onClick(View
arg0) {
201
202                     TambahDataPemilikClass.this.finish();
203                 }
204             });
205             dialog.show();
206         } else {
207             String query = "exec ValidasiLogin '" +
enim + "' ,'" + epassword + "'";
208             Statement stmt = con.createStatement();
209             ResultSet rs =
stmt.executeQuery(query);
210
211             while (rs.next())
212             {
213                 NamaMhs =
rs.getString("Full_Name");
214             }
215         }
216     }
217     catch (Exception ex) {
218         z = "Exceptions "+
ex.getMessage().toString();
219     }
220
221     HashMap<String, String> queryValues = new
HashMap<String, String>();
222     queryValues.put("nim",
editNim.getText().toString());
223     queryValues.put("nama", NamaMhs);
224     queryValues.put("password",
editPassword.getText().toString());
225     controller.insertMahasiswa(queryValues);
226
227     Intent i = new
Intent(TambahDataPemilikClass.this,ValidasiClass.class);
228     startActivity(i);
229     finish();

```

```

230         }
231     }
232     }
233
234     @Override
235     protected String doInBackground(String... params) {
236         if(enim.trim().equals("") ||
epassword.trim().equals(""))
237             z = "Mohon masukkan NIM, Nama dan Password
!!!";
238         else
239             {
240                 try {
241                     Connection con = connectionClass.CONN();
242                     if (con == null) {
243                         final Dialog dialog = new
Dialog(context);
244
245                         dialog.requestWindowFeature(Window.FEATURE_LEFT_ICON);
246
247                         dialog.setFeatureDrawableResource(Window.FEATURE_LEFT_ICON
, R.drawable.logoumy);
248
249                         dialog.setContentview(R.layout.errorkoneksi_layout);
250
251                         dialog.setCancelable(false);
252
253                         dialog.setCanceledOnTouchOutside(false);
254
255                         Button ok = (Button)
dialog.findViewById(R.id.btnok);
256                         Button cancel = (Button)
dialog.findViewById(R.id.btncancel);
257
258                         ok.setOnClickListener(new
OnClickListener() {
259                             public void onClick(View
arg0) {
260                                 TambahDataPemilikClass.this.recreate();
261                             }
262                         });
263
264                         cancel.setOnClickListener(new
OnClickListener() {
265                             public void onClick(View
arg0) {
266                                 TambahDataPemilikClass.this.finish();
267                             }
268                         });
269
270                         dialog.show();
271
272                     } else {

```

```

270             String query = "exec ValidasiLogin '" +
enim + "' ,'" + epassword + "'";
271             Statement stmt =
con.createStatement();
272             ResultSet rs =
stmt.executeQuery(query);
273
274             if (rs.next())
275             {
276                 z = "Login Berhasil";
277                 isSuccess=true;
278             }
279             else if (! rs.next())
280             {
281                 z = "NIM atau Password salah";
282                 isSuccess = false;
283             }
284             }
285         }
286         catch (Exception ex)
287         {
288             isSuccess = false;
289             z = "Exceptions "+
ex.getMessage().toString();
290         }
291     }
292     return z;
293 }
294 }

```

Lampiran 9 Source Code Login java

```

1  package com.example.presensionline;
696.
697.     import java.sql.Connection;
698.     import java.sql.ResultSet;
699.     import java.sql.Statement;
700.     import java.util.HashMap;
701.     import android.app.Activity;
702.     import android.app.AlertDialog;
703.     import android.content.Context;
704.     import android.content.DialogInterface;
705.     import android.content.Intent;
706.     import android.os.AsyncTask;
707.     import android.os.Bundle;
708.     import android.view.View;
709.     import android.view.Window;

```



```
739.         buttonLogin.setOnClickListener(new
View.OnClickListener() {
740.
741.             public void onClick(View v) {
742.
743.                 if
(editNim.getText().toString().trim().equals("") ||
editPassword.getText().toString().trim().equals(""))
744.                     {
745.
android.app.AlertDialog.Builder alert = new
AlertDialog.Builder(TambahDataPemilikClass.this);
746.                         alert.setTitle("Login Gagal");
747.                         alert.setMessage("NIM dan
Password Tidak Boleh Kosong");
748.
749.                         alert.setPositiveButton("Ulangi", new
Dialoginterface.OnClickListener() {
750.                             public void
onClick(Dialoginterface dialog, int id) {
751.                                 dialog.cancel();
752.                             }
753.                         });
754.                         alert.show();
755.                     }
756.                     else if
(editNim.getText().toString().trim().equals(""))
757.                         {
758.
android.app.AlertDialog.Builder alert = new
AlertDialog.Builder(TambahDataPemilikClass.this);
759.                         alert.setTitle("Login Gagal");
760.                         alert.setMessage("NIM Tidak
Boleh Kosong");
761.
```

```
762.
    alert.setPositiveButton("Ulangi", new
    DialogInterface.OnClickListener() {
763.
        public void
    onClick(DialogInterface dialog, int id) {
764.
        dialog.cancel();
765.
        }
766.
    });
767.
    alert.show();
768.
}
769.
770.
    else if
    (editPassword.getText().toString().trim().equals(""))
771.
    {
772.
        android.app.AlertDialog.Builder alert = new
        AlertDialog.Builder(TambahDataPemilikClass.this);
773.
        alert.setTitle("Login Gagal");
774.
        alert.setMessage("Password Tidak
        Boleh Kosong");
775.
776.
        alert.setPositiveButton("Ulangi", new
        DialogInterface.OnClickListener() {
777.
            public void
        onClick(DialogInterface dialog, int id) {
778.
            dialog.cancel();
779.
            }
780.
        });
781.
        alert.show();
782.
    }
783.
    else if
    (editNim.getText().length() != 11)
784.
    {
785.
        android.app.AlertDialog.Builder alert = new
        AlertDialog.Builder(TambahDataPemilikClass.this);
```



```
786.             alert.setTitle("Login Gagal");
787.             alert.setMessage("Format NIM
Salah"+"\\n"+"Mohon Masukkan NIM dengan Benar");
788.
789.             alert.setPositiveButton("Ulangi", new
Dialoginterface.OnClickListener() {
790.                 public void
onClick(Dialoginterface dialog, int id) {
791.                     dialog.cancel();
792.                 }
793.             });
794.             alert.show();
795.         }
796.         else
797.         {
798.             {
799.                 DoLogin doLogin = new
DoLogin();
800.                 doLogin.execute("");
801.             }
802.         }
803.     });
804. }
805.
806.     public void onBackPressed() {
807.         AlertDialog.Builder builder = new
AlertDialog.Builder(this,
AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
808.         builder.setMessage("Apakah Anda Ingin
Keluar?")
809.             .setCancelable(false)
810.             .setPositiveButton("Iya", new
Dialoginterface.OnClickListener() {
811.                 public void
onClick(Dialoginterface dialog, int id) {
```

```
812.     TambahDataPemilikClass.this.finish();
813.         }
814.     })
815.     .setNegativeButton("Tidak", new
816.         DialogInterface.OnClickListener() {
817.             public void
818.             onClick(DialogInterface dialog, int id) {
819.                 dialog.cancel();
820.             }
821.         });
822.     AlertDialog alert = builder.create();
823.     alert.show();
824. }
825.
826.     public class DoLogin extends
827.         AsyncTask<String,String,String>
828.     {
829.         String z = "";
830.         Boolean isSuccess = false;
831.
832.         String enim = editNim.getText().toString();
833.         String epassword =
834.             editPassword.getText().toString();
835.
836.         @Override
837.         protected void onPreExecute() {
838.             pbbar.setVisibility(View.VISIBLE);
839.         }
840.
841.         @Override
842.         protected void onPostExecute(String r) {
843.             pbbar.setVisibility(View.GONE);
```

```
842. Toast.makeText(TambahDataPemilikClass.this,r,Toast.LENGTH_
LONG).show();
843.
844.         if(isSuccess) {
845.             try {
846.                 Connection con =
connectionClass.CONN();
847.                 if (con == null) {
848.                     z = "Mohon sambungkan dengan
jaringan UMY";
849.                 } else {
850.                     String query = "exec
ValidasiLogin '" + enim + "' ,'" + epassword + "'";
851.                     Statement stmt =
con.createStatement();
852.                     ResultSet rs =
stmt.executeQuery(query);
853.
854.                     while (rs.next())
855.                     {
856.                         NamaMhs =
rs.getString("Full_Name");
857.                     }
858.                 }
859.             }
860.             catch (Exception ex) {
861.                 z = "Exceptions "+
ex.getMessage().toString();
862.             }
863.
864.             HashMap<String, String>
queryValues = new HashMap<String, String>();
865.             queryValues.put("nim",
editNim.getText().toString());
866.             queryValues.put("nama", NamaMhs);
```

```
867.             queryValues.put("password",
editPassword.getText().toString());
868.
            controller.insertMahasiswa(queryValues);
869.
870.             Intent i = new
            Intent(TambahDataPemilikClass.this,ValidasiClass.class);
871.             startActivity(i);
872.             finish();
873.         }
874.
875.     }
876.
877.     @Override
878.     protected String doInBackground(String...
params) {
879.         if(enim.trim().equals("")||
epassword.trim().equals(""))
880.             z = "Mohon masukkan NIM, Nama dan
Password !!!";
881.         else
882.         {
883.             try {
884.                 Connection con =
connectionClass.CONN();
885.                 if (con == null) {
886.                     z = "Mohon sambungkan dengan
jaringan UMY";
887.                 } else {
888.                     String query = "exec
ValidasiLogin '" + enim + "' ,'" + epassword + "'";
889.                     Statement stmt =
con.createStatement();
890.                     ResultSet rs =
stmt.executeQuery(query);
891.
892.                     if (rs.next())
```

```

893.         {
894.             z = "Login Berhasil";
895.             isSuccess=true;
896.         }
897.         else if (! rs.next())
898.         {
899.             z = "NIM atau Password
    salah";
900.             isSuccess = false;
901.         }
902.     }
903. }
904. catch (Exception ex)
905. {
906.     isSuccess = false;
907.     z = "Exceptions "+
    ex.getMessage().toString();
908. }
909. }
910. return z;
911. }
912. }
913. }

```

Lampiran 10 Source Code splash screen java

```

1  package com.example.presensionline;
914.
915.     import android.app.Activity;
916.     import android.app.AlertDialog;
917.     import android.app.Dialog;
918.     import android.content.Context;
919.     import android.content.DialogInterface;
920.     import android.content.Intent;
921.     import android.os.Bundle;
922.     import android.os.Handler;

```

```
923.     import android.text.Html;
924.     import android.view.View;
925.     import android.view.Window;
926.     import android.view.View.OnClickListener;
927.     import android.widget.Button;
928.     import android.widget.Toast;
929.     import java.sql.Connection;
930.     import java.util.ArrayList;
931.     import java.util.HashMap;
932.
933.     public class SplashScreen_Class extends Activity {
934.
935.         ConnectionClass connectionClass;
936.         DBController controller = new
            DBController(this);
937.         Boolean isSuccess = false;
938.         final Context context = this;
939.         private final int SPLASH_DISPLAY_LENGTH =
            1000;
940.
941.         @Override
942.         public void onCreate(Bundle icle) {
943.
944.             requestWindowFeature(Window.FEATURE_NO_TITLE);
945.             super.onCreate(icle);
946.             setContentView(R.layout.splash_layout);
947.
948.             new Handler().postDelayed(new Runnable() {
949.                 public void run() {
950.                     android.app.AlertDialog.Builder
951.                         alert = new AlertDialog.Builder(SplashScreen_Class.this,
952.                             AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
953.                     alert.setTitle("Konfirmasi");
954.                     alert.setMessage(Html.fromHtml("Apakah Anda Sudah
955.                         Tersambung Dengan Jaringan UMY?"));
956.                     alert.setCancelable(false);
```

```
953.             alert.setPositiveButton("Sudah", new
Dialoginterface.OnClickListener() {
954.                 public void
onClick(Dialoginterface dialog, int id) {
955.                     dialog.cancel();
956.                     connectionClass = new
ConnectionClass();
957.                 try {
958.                     Connection con =
connectionClass.CONN();
959.                     if (con == null) {
960.                         String z =
"Mohon sambungkan dengan jaringan UMY";
961.                         Toast.makeText(SplashScreen_Class.this, z,
962. Toast.LENGTH_SHORT).show();
963.                         final Dialog
dialog1 = new Dialog(context);
964.                         dialog1.setContentView(R.layout.errorkoneksi_layout)
;
965.                         dialog1.setTitle("Terjadi Kesalahan Dengan
Jaringan");
966.                         dialog1.setCancelable(false);
967.                         dialog1.setCanceledOnTouchOutside(false);
968.                         Button ok =
(Button) dialog1.findViewById(R.id.btnok);
969.                         Button cancel =
(Button) dialog1.findViewById(R.id.btncancel);
970.                         ok.setOnClickListener(new OnClickListener() {
971.
```

```
972.     public void onClick(View arg0) {
973.         SplashScreen_Class.this.recreate();
974.     }
975.     });
976.
977.     cancel.setOnClickListener(new OnClickListener() {
978.
979.     public void onClick(View arg0) {
980.         SplashScreen_Class.this.finish();
981.     }
982.     });
983.
984.         dialog1.show();
985.     }
986.     else
987.     {
988.
989.         ArrayList<HashMap<String, String>> dataCari =
990.         controller.getDataMahasiswa(1);
991.         if
992.         Intent
993.         mainIntent = new
994.         Intent(SplashScreen_Class.this, TambahDataPemilikClass.clas
995.         s);
996.         SplashScreen_Class.this.startActivity(mainIntent);
997.         SplashScreen_Class.this.finish();
998.     }
999.     else
1000.    {
```



```
996.                                     Intent
    mainIntent = new
    Intent(SplashScreen_Class.this,LoginClass.class);
997.
    SplashScreen_Class.this.startActivity(mainIntent);
998.
    SplashScreen_Class.this.finish();
999.                                     }
1000.                                     }
1001.                                     }
1002.                                     catch (Exception e) {
1003.                                     e.getMessage();
1004.                                     }
1005.
1006.                                     }
1007.                                     });
1008.                                     alert.setNegativeButton("Belum", new
    DialogInterface.OnClickListener() {
1009.                                     public void
    onClick(DialogInterface dialog, int id) {
1010.                                     dialog.cancel();
1011.
    SplashScreen_Class.this.finish();
1012.
    }
1013.                                     });
1014.                                     });
1015.
1016.                                     alert.create();
1017.                                     alert.show();
1018.                                     }
1019.                                     }, SPLASH_DISPLAY_LENGTH);
1020.                                     }
1021.                                     }
```

Lampiran 11 Source Code Validasi Java

```
1 package com.example.presensionline;
1022.
1023.     import android.app.Activity;
1024.     import android.app.AlertDialog;
1025.     import android.app.Dialog;
1026.     import android.content.Context;
1027.     import android.content.DialogInterface;
1028.     import android.os.AsyncTask;
1029.     import android.os.Bundle;
1030.     import android.text.Html;
1031.     import android.view.View;
1032.     import android.view.Window;
1033.     import android.view.View.OnClickListener;
1034.     import android.widget.Button;
1035.     import android.widget.EditText;
1036.     import android.widget.ListView;
1037.     import android.widget.ProgressBar;
1038.     import android.widget.TextView;
1039.     import android.widget.Toast;
1040.     import java.sql.Connection;
1041.     import java.sql.PreparedStatement;
1042.     import java.sql.ResultSet;
1043.     import java.sql.Statement;
1044.     import java.util.ArrayList;
1045.     import java.util.HashMap;
1046.
1047.
1048.     public class ValidasiClass extends Activity {
1049.
1050.         final Context context = this;
1051.         DBController controller = new
            DBController(this);
1052.         ConnectionClass connectionClass;
1053.         EditText edtTOKEN;
1054.         TextView user;
1055.         Button btnSubmit;
```

```
1056.         ProgressBar pbbar;
1057.         ListView lstpro;
1058.         String proid, NIM, Nama;
1059.         String z = "";
1060.
1061.
1062.         @Override
1063.         protected void onCreate(Bundle
            savedInstanceState) {
1064.             super.onCreate(savedInstanceState);
1065.             setContentView(R.layout.validasi_layout);
1066.
1067.             connectionClass = new ConnectionClass();
1068.             edtTOKEN = (EditText)
                findViewById(R.id.edtoken);
1069.             btnSubmit = (Button)
                findViewById(R.id.btnadd);
1070.             pbbar = (ProgressBar)
                findViewById(R.id.pbbar);
1071.             pbbar.setVisibility(View.GONE);
1072.             proid = "";
1073.             user =
                (TextView) findViewById(R.id.username);
1074.             ArrayList<HashMap<String, String>> dataCari
                = controller.getDataMahasiswa(1);
1075.             NIM = dataCari.get(0).get("nim");
1076.
                user.setText(dataCari.get(0).get("nama"));
1077.             btnSubmit.setOnClickListener(new
                View.OnClickListener() {
1078.                 public void onClick(View v) {
1079.                     CekToken CK = new CekToken();
1080.                     CK.execute("");
1081.                 }
1082.             });
1083.         }
1084.
```

```
1085.         public void onBackPressed() {
1086.             AlertDialog.Builder builder = new
                AlertDialog.Builder(this,
                AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
1087.             builder.setMessage("Apakah Anda Ingin
                Keluar?")
1088.                 .setCancelable(false)
1089.                 .setPositiveButton("Iya", new
                DialogInterface.OnClickListener() {
1090.                     public void
                onClick(DialogInterface dialog, int id) {
1091.                         ValidasiClass.this.finish();
1092.                     }
1093.                 })
1094.                 .setNegativeButton("Tidak", new
                DialogInterface.OnClickListener() {
1095.                     public void
                onClick(DialogInterface dialog, int id) {
1096.                         dialog.cancel();
1097.                     }
1098.                 });
1099.             AlertDialog alert = builder.create();
1100.             alert.show();
1101.         }
1102.
1103.         public class CekToken extends
                AsyncTask<String,String,String>
1104.         {
1105.             String z = "";
1106.             Boolean isSuccess = false;
1107.
1108.             @Override
1109.             protected void onPreExecute() {
1110.                 pbbar.setVisibility(View.VISIBLE);
1111.             }
1112.
1113.             @Override
```

```
1114.         protected void onPostExecute(String r) {
1115.             pbbar.setVisibility(View.GONE);
1116.
1117.             if(isSuccess) {
1118.                 final Dialog dialog = new
Dialog(context);
1119.                 dialog.requestWindowFeature(Window.FEATURE_LEFT_ICON
);
1120.                 dialog.setFeatureDrawableResource(Window.FEATURE_LEF
T_ICON, R.drawable.logoumy);
1121.                 dialog.setContentView(R.layout.konfirmasi_layout);
1122.                 dialog.setTitle("Detail
Presensi");
1123.                 dialog.setCancelable(false);
1124.
dialog.setCanceledOnTouchOutside(false);
1125.                 TextView nmMk, klsmk, dsn, ssmk,
hrmk, tglmk, mulai, toleransi;
1126.                 nmMk =
(TextView)dialog.findViewById(R.id.tvNamaMK);
1127.                 klsmk =
(TextView)dialog.findViewById(R.id.tvNamaKl);
1128.                 dsn =
(TextView)dialog.findViewById(R.id.tvNamaDs);
1129.                 ssmk =
(TextView)dialog.findViewById(R.id.tvNamaSs);
1130.                 hrmk =
(TextView)dialog.findViewById(R.id.tvNamaHr);
1131.                 tglmk =
(TextView)dialog.findViewById(R.id.tvNamaTgl);
1132.                 mulai =
(TextView)dialog.findViewById(R.id.tvJmMulai);
1133.                 toleransi =
(TextView)dialog.findViewById(R.id.tvmt);
```

```
1134.             Button ok = (Button)
                dialog.findViewById(R.id.btnok);
1135.             Button cancel = (Button)
                dialog.findViewById(R.id.btncancel);
1136.             try
1137.             {
1138.                 Connection con =
                connectionClass.CONN();
1139.                 String query = "exec
                KetersediaanToken '"+edtTOKEN.getText().toString()+"'";
1140.                 Statement stmt =
                con.createStatement();
1141.                 ResultSet rs =
                stmt.executeQuery(query);
1142.                 while(rs.next())
1143.                 {
1144.                     nmMk.setText(
                rs.getString(1));
1145.                     klsmk.setText(
                rs.getString(2));
1146.                     dsn.setText(
                rs.getString(3));
1147.                     ssmk.setText(
                rs.getString(4));
1148.                     hrmk.setText
                (rs.getString(5));
1149.                     tglmk.setText(
                rs.getString(6));
1150.
                mulai.setText(rs.getString(7));
1151.                     toleransi.setText
                (rs.getString(8));
1152.                 }
1153.             }
1154.             catch (Exception e) {
1155.                 }
```

```
1156.                final String a =
                    nmMk.getText().toString();
1157.                ok.setOnClickListener(new
                    OnClickListener() {
1158.                    public void
                    onClick(View arg0) {
1159.                        android.app.AlertDialog.Builder alert = new
                    AlertDialog.Builder(ValidasiClass.this,AlertDialog.THEME_D
                    EVICE_DEFAULT_LIGHT);
1160.                        alert.setTitle("Konfirmasi Presensi");
1161.                        alert.setMessage("Apakah Anda Yakin Akan Melakukan
                    Presensi Pada Mata Kuliah "+ a);
1162.                        alert.setCancelable(false);
1163.                        alert.setPositiveButton("Iya", new
                    DialogInterface.OnClickListener() {
1164.                            public void
                            onClick(DialogInterface dialog, int id) {
1165.                                validasipresensi vp = new validasipresensi();
1166.                                vp.execute("");
1167.                            }
1168.                        })
1169.
1170.                        .setNegativeButton("Tidak", new
                    DialogInterface.OnClickListener() {
1171.                            public void
                            onClick(DialogInterface dialog, int id) {
1172.                                dialog.cancel();
1173.                            }
                    }
```

```
1174.                                     });
1175.                                     alert.show();
1176.                                     dialog.cancel();
1177.                                     }
1178.                                 });
1179.
1180.                                 cancel.setOnClickListener(new
    OnClickListener() {
1181.                                     public void
    onClick(View arg0) {
1182.                                         dialog.cancel();
1183.
1184.                                     }
1185.                                 });
1186.
1187.                                 dialog.show();
1188.                                 }
1189.                                 else
1190.                                 {
1191.                                     android.app.AlertDialog.Builder
    alert = new AlertDialog.Builder(ValidasiClass.this,
    AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
1192.                                     alert.setTitle("Validasi Token
    Gagal");
1193.                                     alert.setMessage("Token Tidak
    Terdaftar pada hari ini!!!");
1194.                                     alert.setCancelable(false);
1195.                                     alert.setPositiveButton("Ulangi",
    new DialogInterface.OnClickListener() {
1196.                                         public void
    onClick(DialogInterface dialog, int id) {
1197.                                             dialog.cancel();
1198.                                         }
1199.                                     });
1200.                                     alert.show();
1201.                                 }
1202.                                 }
```



```
1203.
1204.         @Override
1205.         protected String doInBackground(String...
           params) {
1206.             if(edtTOKEN.getText().toString().trim().equals(""))
1207.                 z = "Mohon Masukkan TOKEN !!!";
1208.             else
1209.             {
1210.                 try {
1211.                     Connection con =
           connectionClass.CONN();
1212.                     if (con == null) {
1213.                         z = "Mohon sambungkan dengan
           jaringan UMY";
1214.                     } else {
1215.                         String query = "exec
           KetersediaanToken '"+edtTOKEN.getText().toString()+"'";
1216.                         Statement stmt =
           con.createStatement();
1217.                         ResultSet rs =
           stmt.executeQuery(query);
1218.
1219.                         if(rs.next())
1220.                         {
1221.                             isSuccess=true;
1222.                         }
1223.                         else
1224.                         {
1225.                             isSuccess = false;
1226.                         }
1227.                     }
1228.                 }
1229.                 catch (Exception ex)
1230.                 {
1231.                     isSuccess = false;
1232.                     z = "Exceptions";
```

```
1233.         }
1234.     }
1235.         return z;
1236.     }
1237. }
1238.
1239.     public class validasipresensi extends
        AsyncTask<String,String,String>
1240.     {
1241.         String token = edtTOKEN.getText().toString();
1242.         String z = "";
1243.         Boolean isSuccess = false;
1244.
1245.         @Override
1246.         protected void onPreExecute() {
1247.             pbar.setVisibility(View.VISIBLE);
1248.         }
1249.
1250.         @Override
1251.         protected void onPostExecute(String r) {
1252.             pbar.setVisibility(View.GONE);
1253.
1254.             if(isSuccess) {
1255.                 CekKRS Cek = new CekKRS();
1256.                 Cek.execute("");
1257.             }
1258.             else
1259.             {
1260.                 android.app.AlertDialog.Builder
                alert = new AlertDialog.Builder(ValidasiClass.this,
                AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
1261.                 alert.setTitle("Presensi Gagal");
1262.                 alert.setMessage("Anda sudah
                melakukan presensi pada mata kuliah ini sebelumnya...");
1263.                 alert.setCancelable(false);
1264.                 alert.setPositiveButton("Oke", new
                DialogInterface.OnClickListener() {
```

```
1265.         public void
           onClick(Dialoginterface dialog, int id) {
1266.             dialog.cancel();
1267.         }
1268.     });alert.show();
1269.     }
1270. }
1271.
1272.     @Override
1273.     protected String doInBackground(String...
           params) {
1274.         try {
1275.             Connection con =
           connectionClass.CONN();
1276.             if (con == null) {
1277.                 final Dialog dialog = new
           Dialog(context);
1278.
           dialog.requestWindowFeature(Window.FEATURE_LEFT_ICON
           );
1279.
           dialog.setFeatureDrawableResource(Window.FEATURE_LEF
           T_ICON, R.drawable.logoumy);
1280.
           dialog.setContentview(R.layout.errorkoneksi_layout);
1281.             dialog.setTitle("Terjadi
           Kesalahan Dengan Jaringan");
1282.             dialog.setCancelable(false);
1283.
           dialog.setCanceledOnTouchOutside(false);
1284.             Button ok = (Button)
           dialog.findViewById(R.id.btnok);
1285.             Button cancel = (Button)
           dialog.findViewById(R.id.btncancel);
1286.
           ok.setOnClickListener(new OnClickListener() {
```

```
1287.                                     public void
      onClick(View arg0) {
1288.
          ValidasiClass.this.recreate();
1289.                                     }
1290.                                     });
1291.
1292.
          cancel.setOnClickListener(new OnClickListener() {
1293.
1294.                                     public void
      onClick(View arg0) {
1295.
          ValidasiClass.this.finish();
1296.                                     }
1297.                                     });
1298.                                     dialog.show();
1299.                                     } else {
1300.                                     String query = "exec
      spValidasipresensi '"+NIM+"', '"+token+"'";
1301.                                     Statement stmt =
      con.createStatement();
1302.                                     ResultSet rs =
      stmt.executeQuery(query);
1303.                                     if(rs.next())
1304.                                     {
1305.                                     isSuccess=false;
1306.                                     }
1307.                                     else
1308.                                     {
1309.                                     z = "";
1310.                                     isSuccess = true;
1311.                                     }
1312.                                     }
1313.                                     }
1314.                                     catch (Exception ex)
1315.                                     {
```

```
1316.             isSuccess = false;
1317.             z = "Exceptions" +
    ex.getMessage().toString();
1318.             }
1319.
1320.             return z;
1321.         }
1322.     }
1323.
1324.     public class CekKRS extends
    AsyncTask<String,String,String>
1325.     {
1326.         String z = "";
1327.         Boolean isSuccess = false;
1328.         Boolean isKoneksiSuccess = false;
1329.         String token =
    edtTOKEN.getText().toString();
1330.
1331.         @Override
1332.         protected void onPreExecute() {
1333.             pbbar.setVisibility(View.VISIBLE);
1334.         }
1335.
1336.         @Override
1337.         protected void onPostExecute(String r) {
1338.             pbbar.setVisibility(View.GONE);
1339.
1340.             if(isSuccess) {
1341.                 validasiKelas vk = new
    validasiKelas();
1342.                 vk.execute("");
1343.             }
1344.             else
1345.             {
1346.                 String namaMk = "a";
1347.                 try
1348.                 {
```

```
1349.                Connection con =
    connectionClass.CONN();
1350.                String query = "exec
    kuliahToken '"+token+"'";
1351.                Statement stmt =
    con.createStatement();
1352.                ResultSet rs =
    stmt.executeQuery(query);
1353.
1354.                while(rs.next())
1355.                {
1356.                    namaMk = rs.getString(1);
1357.                }
1358.
1359.                android.app.AlertDialog.Builder
    alert = new AlertDialog.Builder(ValidasiClass.this,
    AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
1360.                alert.setTitle("Presensi
    Gagal");
1361.
    alert.setMessage(Html.fromHtml("<font
    color='"+"#B0272B"+">"+"ANDA BELUM TERDAFTAR"+"</font>"+
    Untuk Mata Kuliah "+<b>"+ " "+namaMk+"</b>"+<br>"+
1362.                "Silahkan Melakukan Key-in
    Terlebih Dahulu Untuk Bisa Melakukan Presensi Pada
    Matakuliah "+<b>"+ " "+namaMk+"</b>"));
1363.                alert.setCancelable(false);
1364.                alert.setPositiveButton("OK",
    new DialogInterface.OnClickListener() {
1365.                    public void
    onClick(DialogInterface dialog, int id) {
1366.                        dialog.cancel();
1367.                    }
1368.                });
1369.                alert.show();
1370.            }
1371.            catch (Exception e) {
```

```
1372.                z = "Exceptions" +
    e.getMessage().toString();
1373.
    android.app.AlertDialog.Builder alert = new
    AlertDialog.Builder(ValidasiClass.this,
    AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
1374.                alert.setTitle("Error...");
1375.                alert.setMessage("Terjadi
    kesalahan dengan sistem. Silahkan menghubungi BSI");
1376.                alert.setCancelable(false);
1377.
    alert.setPositiveButton("Ulangi", new
    DialogInterface.OnClickListener() {
1378.                public void
    onClick(DialogInterface dialog, int id) {
1379.                dialog.cancel();
1380.                }
1381.                });
1382.                alert.show();
1383.                }
1384.            }
1385.        }
1386.
1387.        @Override
1388.        protected String doInBackground(String...
    params) {
1389.                if(token.trim().equals(""))
1390.                z = "Mohon masukkan Token";
1391.                else
1392.                {
1393.                try {
1394.                Connection con =
    connectionClass.CONN();
1395.                if (con == null) {
1396.                final Dialog dialog = new
    Dialog(context);
```

```
1397.         dialog.requestWindowFeature(Window.FEATURE_LEFT_ICON
1398.         );
1399.         dialog.setFeatureDrawableResource(Window.FEATURE_LEF
1400.         T_ICON, R.drawable.logoumy);
1401.         dialog.setContentview(R.layout.errorkoneksi_layout);
1402.         dialog.setTitle("Terjadi
1403.         Kesalahan Dengan Jaringan");
1404.         dialog.setCancelable(false);
1405.         dialog.setCanceledOnTouchOutside(false);
1406.         Button ok = (Button)
1407.         dialog.findViewById(R.id.btnok);
1408.         Button cancel = (Button)
1409.         dialog.findViewById(R.id.btncancel);
1410.         ok.setOnClickListener(new OnClickListener() {
1411.             public void
1412.             onClick(View arg0) {
1413.                 ValidasiClass.this.recreate();
1414.             }
1415.         });
1416.         cancel.setOnClickListener(new OnClickListener() {
1417.             public void
1418.             onClick(View arg0) {
1419.                 ValidasiClass.this.finish();
1420.             }
1421.         });
1422.         dialog.show();
1423.     } else {
```



```
1419.                                     String query = "exec CekKRS
      '"+ NIM +"' , '"+token+"'";
1420.                                     Statement stmt =
      con.createStatement();
1421.                                     ResultSet rs =
      stmt.executeQuery(query);
1422.
1423.                                     if(rs.next())
1424.                                     {
1425.                                         isSuccess=true;
1426.                                     }
1427.                                     else
1428.                                     {
1429.                                         isSuccess = false;
1430.                                     }
1431.                                     }
1432.                                     }
1433.                                     catch (Exception ex)
1434.                                     {
1435.                                         z = "Exceptions" +
      ex.getMessage().toString();
1436.                                     }
1437.                                     }
1438.                                     return z;
1439.                                     }
1440.                                     }
1441.
1442.     public class validasiKelas extends
      AsyncTask<String,String,String>
1443.     {
1444.         String z = "";
1445.         Boolean isSuccess = false;
1446.         Boolean isKoneksiSuccess = false;
1447.         String token =
      edtTOKEN.getText().toString();
1448.
1449.         @Override
```

```
1450.         protected void onPreExecute() {
1451.             pbbar.setVisibility(View.VISIBLE);
1452.         }
1453.
1454.         @Override
1455.         protected void onPostExecute(String r) {
1456.             pbbar.setVisibility(View.GONE);
1457.
1458.             Toast.makeText(ValidasiClass.this, r, Toast.LENGTH_SHORT).show();
1459.
1460.             if (isSuccess) {
1461.                 insertmhstepatwaktu imw = new
1462.                 insertmhstepatwaktu();
1463.                 imw.execute("");
1464.             }
1465.             else
1466.             {
1467.                 android.app.AlertDialog.Builder
1468.                 alert = new AlertDialog.Builder(ValidasiClass.this,
1469.                 AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
1470.                 alert.setTitle("Presensi Gagal");
1471.                 alert.setMessage(Html.fromHtml("Anda "+"<b>"+"Tidak
1472.                 Terdaftar"+"</b>"+" dalam kelas ini"));
1473.                 alert.setCancelable(false);
1474.                 alert.setPositiveButton("Ulangi",
1475.                 new DialogInterface.OnClickListener() {
1476.                     public void
1477.                     onClick(DialogInterface dialog, int id) {
1478.                         dialog.cancel();
1479.                     }
1480.                 });
1481.                 alert.show();
1482.             }
1483.         }
1484.     }
```

```
1478.         @Override
1479.         protected String doInBackground(String...
           params) {
1480.             if(token.trim().equals(""))
1481.                 z = "Mohon masukkan Token";
1482.             else
1483.                 {
1484.                     user =
                       (TextView)findViewById(R.id.username);
1485.                     Bundle b = getIntent().getExtras();
1486.
1487.                     try {
1488.                         Connection con =
                           connectionClass.CONN();
1489.                         if (con == null) {
1490.                             final Dialog dialog = new
                               Dialog(context);
1491.
                               dialog.requestWindowFeature(Window.FEATURE_LEFT_ICON
                               );
1492.
                               dialog.setFeatureDrawableResource(Window.FEATURE_LEF
                               T_ICON, R.drawable.logoumy);
1493.
                               dialog.setContentview(R.layout.errorkoneksi_layout);
1494.
                               dialog.setTitle("Terjadi
                                   Kesalahan Dengan Jaringan");
1495.
                               dialog.setCancelable(false);
1496.
                               dialog.setCanceledOnTouchOutside(false);
1497.
                               Button ok = (Button)
                                   dialog.findViewById(R.id.btnok);
1498.
                               Button cancel = (Button)
                                   dialog.findViewById(R.id.btncancel);
1499.
1500.
                                   ok.setOnClickListener(new OnClickListener() {
```

```
1501.
1502.                                     public void
      onClick(View arg0) {
1503.
      ValidasiClass.this.recreate();
1504.                                     }
1505.                                     });
1506.
1507.
      cancel.setOnClickListener(new OnClickListener() {
1508.
1509.                                     public void
      onClick(View arg0) {
1510.
      ValidasiClass.this.finish();
1511.                                     }
1512.                                     });
1513.
1514.                                     dialog.show();
1515.                                     } else {
1516.                                     String query = "exec
      cekkelas '"+ NIM +"', '"+token+"'";
1517.                                     Statement stmt =
      con.createStatement();
1518.                                     ResultSet rs =
      stmt.executeQuery(query);
1519.
1520.                                     if(rs.next())
1521.                                     {
1522.                                     isSuccess=true;
1523.                                     }
1524.                                     else
1525.                                     {
1526.                                     isSuccess = false;
1527.                                     }
1528.                                     }
1529.                                     }
```

```

1530.             catch (Exception ex)
1531.             {
1532.                 isSuccess = false;
1533.                 z = "Exceptions" +
ex.getMessage().toString();
1534.             }
1535.         }
1536.         return z;
1537.     }
1538. }
1539.
1540.     public class insertmhstepatwaktu extends
AsyncTask<String,String,String>
1541.     {
1542.         String z = "";
1543.         Boolean isSuccess = false;
1544.         Boolean isKoneksiSuccess = false;
1545.         String token =
edtTOKEN.getText().toString();
1546.         String nim, nama,namamk,waktu, status;
1547.
1548.         @Override
1549.         protected void onPreExecute() {
1550.             pbbar.setVisibility(View.VISIBLE);
1551.         }
1552.
1553.         @Override
1554.         protected void onPostExecute(String r) {
1555.             pbbar.setVisibility(View.GONE);
1556.
1557.             if(isSuccess) {
1558.                 try
1559.                 {
1560.                     Connection con =
connectionClass.CONN();
1561.                     String query = "exec
BerhasilPresensi '"+ NIM +'";

```

```
1562.                Statement stmt =
    con.createStatement();
1563.                ResultSet rs =
    stmt.executeQuery(query);
1564.
1565.                while (rs.next())
1566.                {
1567.                    nim = rs.getString(1);
1568.                    nama = rs.getString(2);
1569.                    namamk = rs.getString(3);
1570.                    waktu = rs.getString(4);
1571.                    status = rs.getString(10);
1572.                }
1573.
1574.                }
1575.                catch (Exception e) {
1576.                    }
1577.
1578.                android.app.AlertDialog.Builder
    alert = new AlertDialog.Builder(ValidasiClass.this,
    AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
1579.                alert.setTitle("Presensi
    Berhasil");
1580.
    alert.setMessage(Html.fromHtml("Selamat, Anda berhasil
    melakukan presensi pada mata kuliah
    "+"<b>"+namamk+"</b>"+<br/>" +" dengan status "+"<b>"+
    +status+"</b>"));
1581.                alert.setCancelable(false);
1582.
1583.                alert.setPositiveButton("Keluar",
    new DialogInterface.OnClickListener() {
1584.                    public void
    onClick(DialogInterface dialog, int id) {
1585.                        dialog.cancel();
1586.                    }
1587.                });
```

```
1588.             alert.show();
1589.             }
1590.             else
1591.             {
1592.                 insertmhsterlambat imt = new
insertmhsterlambat();
1593.                 imt.execute("");
1594.             }
1595.         }
1596.
1597.         @Override
1598.         protected String doInBackground(String...
params) {
1599.             if(token.trim().equals(""))
1600.                 z = "Mohon masukkan Token";
1601.             else
1602.             {
1603.                 user =
(TextView)findViewById(R.id.username);
1604.                 Bundle b = getIntent().getExtras();
1605.                 try {
1606.                     Connection con =
connectionClass.CONN();
1607.                     if (con == null) {
1608.                         final Dialog dialog = new
Dialog(context);
1609.
dialog.requestWindowFeature(Window.FEATURE_LEFT_ICON
);
1610.
dialog.setFeatureDrawableResource(Window.FEATURE_LEF
T_ICON, R.drawable.logoumy);
1611.
dialog.setContentview(R.layout.errorkoneksi_layout);
1612.
dialog.setTitle("Terjadi
Kesalahan Dengan Jaringan");
1613.
dialog.setCancelable(false);
```

```
1614.         dialog.setCanceledOnTouchOutside(false);
1615.             Button ok = (Button)
                dialog.findViewById(R.id.btnok);
1616.             Button cancel = (Button)
                dialog.findViewById(R.id.btncancel);
1617.
1618.
                ok.setOnClickListener(new OnClickListener() {
1619.
1620.                 public void
                    onClick(View arg0) {
1621.
1622.                     ValidasiClass.this.recreate();
1623.                 }
1624.             });
1625.
                cancel.setOnClickListener(new OnClickListener() {
1626.
1627.                 public void
                    onClick(View arg0) {
1628.
1629.                     ValidasiClass.this.finish();
1630.                 }
1631.             });
1632.             dialog.show();
1633.         } else {
1634.             String query = "exec
                InsertMhsTepatWaktu '"+NIM+"', '"+token+"'";
1635.             PreparedStatement
                preparedStatement = con.prepareStatement(query);
1636.             int jml =
                preparedStatement.executeUpdate();
1637.
1638.             if (jml ==1)
```



```
1639.         {
1640.
1641.             isSuccess=true;
1642.         }
1643.         else
1644.         {
1645.             isSuccess=false;
1646.         }
1647.     }
1648. }
1649.     catch (Exception ex)
1650.     {
1651.         z = "Exceptions" +
1652.           ex.getMessage().toString();
1653.     }
1654.     return z;
1655. }
1656. }
1657.
1658.     public class insertmhsterlambat extends
1659.       AsyncTask<String,String,String>
1660.     {
1661.         String z = "";
1662.         Boolean isSuccess = false;
1663.         Boolean isKoneksiSuccess = false;
1664.         String token =
1665.           edtTOKEN.getText().toString();
1666.         String nim, nama,namamk,waktu, status;
1667.
1668.         @Override
1669.         protected void onPreExecute() {
1670.             pbbbar.setVisibility(View.VISIBLE);
1671.         }
1672.
1673.         @Override
1674.         protected void onPostExecute(String r) {
```

```
1673.                pbbar.setVisibility(View.GONE);
1674.
1675.                if(isSuccess) {
1676.                    try
1677.                    {
1678.                        Connection con =
1679.                            connectionClass.CONN();
1680.                        String query = "exec
1681.                            BerhasilPresensi '"+ NIM +"'";
1682.                        Statement stmt =
1683.                            con.createStatement();
1684.                        ResultSet rs =
1685.                            stmt.executeQuery(query);
1686.                        while (rs.next())
1687.                        {
1688.                            nim = rs.getString(1);
1689.                            nama = rs.getString(2);
1690.                            namamk = rs.getString(3);
1691.                            waktu = rs.getString(4);
1692.                            status = rs.getString(10);
1693.                        }
1694.                    }
1695.                    catch (Exception e) {
1696.                        // TODO: handle
1697.                        exception
1698.                    }
1699.                    android.app.AlertDialog.Builder
1700.                    alert = new AlertDialog.Builder(ValidasiClass.this,
1701.                        AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
1702.                    alert.setTitle("Presensi
1703.                        Berhasil");
1704.                    alert.setMessage(Html.fromHtml("Selamat, Anda berhasil
1705.                        melakukan presensi pada mata kuliah
1706.                        "+"<b>"+namamk+"</b>"+<br/>" + " dengan status "+"<b>"
1707.                        +status+"</b>"));
```

```
1698.
1699.             alert.setCancelable(false);
1700.
1701.             alert.setPositiveButton("Keluar",
    new DialogInterface.OnClickListener() {
1702.                 public void
    onClick(DialogInterface dialog, int id) {
1703.                     dialog.cancel();
1704.                 }
1705.             });
1706.             alert.show();
1707.         }
1708.         else
1709.         {
1710.             String namaMk = "a";
1711.             try
1712.             {
1713.                 Connection con =
    connectionClass.CONN();
1714.                 String query = "exec
    kuliahToken '"+token+"'";
1715.                 Statement stmt =
    con.createStatement();
1716.                 ResultSet rs =
    stmt.executeQuery(query);
1717.
1718.                 while(rs.next())
1719.                 {
1720.                     namaMk = rs.getString(1);
1721.                 }
1722.
1723.                 android.app.AlertDialog.Builder
    alert = new AlertDialog.Builder(ValidasiClass.this,
    AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
1724.                 alert.setTitle("Presensi
    Gagal");
```

```

1725.
    alert.setMessage(Html.fromHtml("<font
    color="+"#B0272B"+">"+ "Sesi Untuk Mata Kuliah"+ "</font>"+
    "+<b>"+ " "+namaMk+"</b>"+ "\n"+
1726.
        "Telah Ditutup
    "+<b>"+ "\n"+ "Anda Harus Datang Tepat Waktu Untuk Dapat
    Melakukan Presensi Untuk Mata Kuliah "+namaMk+"</b>"));
1727.
        alert.setCancelable(false);
1728.
        alert.setPositiveButton("OK",
    new DialogInterface.OnClickListener() {
1729.
            public void
    onClick(DialogInterface dialog, int id) {
1730.
                dialog.cancel();
1731.
            }
1732.
        });
1733.
        alert.show();
1734.
    }
1735.
    catch (Exception e) {
1736.
        z = "Exceptions" +
    e.getMessage().toString();
1737.
        android.app.AlertDialog.Builder alert = new
    AlertDialog.Builder(ValidasiClass.this,
    AlertDialog.THEME_DEVICE_DEFAULT_LIGHT);
1738.
        alert.setTitle("Error...");
1739.
        alert.setMessage("Terjadi
    kesalahan dengan sistem. Silahkan menghubungi BSI");
1740.
        alert.setCancelable(false);
1741.
        alert.setPositiveButton("Ulangi", new
    DialogInterface.OnClickListener() {
1742.
            public void
    onClick(DialogInterface dialog, int id) {
1743.
                dialog.cancel();
1744.
            }
1745.
        });
1746.
        alert.show();

```



```

1778.             z = "Exceptions" +
           ex.getMessage().toString();
1779.             }
1780.         }
1781.         return z;
1782.     }
1783. }
1784. }

```

Lampiran 12 Source Code Androidmanifest

```

1  <?xml version="1.0" encoding="utf-8"?>
1785.  <manifest
           xmlns:android="http://schemas.android.com/apk/res/android"
1786.      package="com.example.presensionline"
1787.      android:versionCode="1"
1788.      android:versionName="1.0" >
1789.
1790.      <uses-permission
           android:name="android.permission.INTERNET" />
1791.      <uses-permission
           android:name="android.permission.ACCESS_NETWORK_STATE" />
1792.      <uses-permission
           android:name="android.permission.ACCESS_WIFI_STATE" />
1793.
1794.      <uses-sdk
1795.          android:minSdkVersion="17"
1796.          android:targetSdkVersion="21" />
1797.
1798.      <application
1799.          android:allowBackup="true"
1800.          android:icon="@drawable/icon_presensi"
1801.          android:label="@string/app_name"
1802.          android:theme="@style/AppTheme" >
1803.          <activity

```

```
1804.         android:name="com.example.presensionline.SplashScreen_Clas
           s"
1805.             android:icon="@drawable/icon_presensi"
1806.             android:label="@string/label_login"
1807.             android:screenOrientation="portrait" >
1808.             <intent-filter>
1809.                 <action
           android:name="android.intent.action.MAIN" />
1810.
1811.                 <category
           android:name="android.intent.category.LAUNCHER" />
1812.             </intent-filter>
1813.         </activity>
1814.
1815.         <activity
1816.             android:label="@string/label_login"
1817.             android:icon="@drawable/logoumy"
1818.             android:name=".TambahDataPemilikClass"
1819.             android:screenOrientation="portrait" />
1820.
1821.         <activity
1822.             android:name=".LoginClass"
1823.             android:label="@string/label_login"
1824.             android:icon="@drawable/logoumy"
1825.             android:screenOrientation="portrait" />
1826.
1827.         <activity
1828.
           android:label="@string/label_login"
1829.             android:icon="@drawable/logoumy"
1830.             android:name=".SubmitClass"
1831.             android:screenOrientation="portrait"
1832.             />
1833.
1834.         <activity
1835.             android:label="@string/label_login"
```

```
1836.             android:icon="@drawable/logoumy"  
1837.             android:name=".ValidasiClass"  
1838.             android:screenOrientation="portrait"/>  
1839.         </application>  
1840.     </manifest>
```