

## BAB IV

### HASIL DAN PEMBAHASAN

Dalam bab ini, penulis akan menguraikan dan menjelaskan beberapa hasil pengujian dari hasil penelitian tugas akhir ini. Pengujian yang dilakukan meliputi pengujian *software* yang berupa pengujian *web server* dan kinerja keseluruhan aplikasi saat dijalankan, serta kemampuan aplikasi pada beberapa versi android.

#### 4.1 Pengujian Web Server

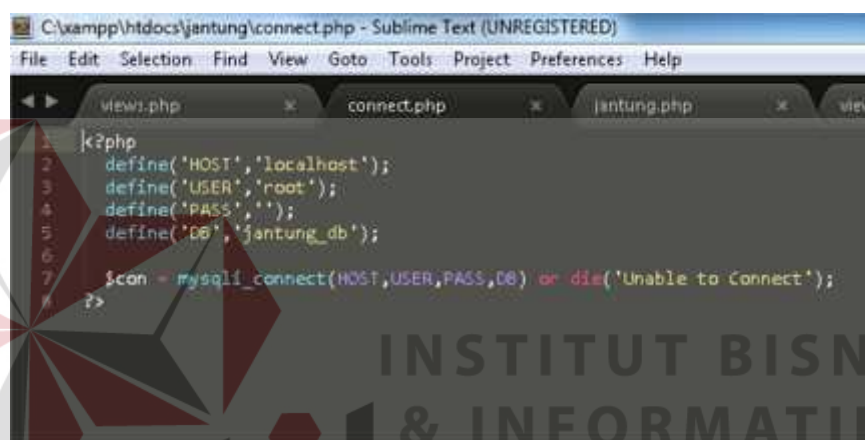
Seperti yang telah dijelaskan pada bab 3, dalam pembuatan aplikasi Sinyal Jantung, dibutuhkan sebuah wadah untuk menampung keseluruhan data pada aplikasi.

Pada setiap tipe android memiliki media penyimpanan dengan kapasitas yang berbeda – beda dan cenderung minim, sedangkan data aplikasi jantung membutuhkan kapasitas yang cukup besar untuk menampung dari para pasien yang menggunakan aplikasi ini. maka dibutuhkan sebuah *web server* untuk penyimpanan data sinyal auskultasi jantung. Maka dibutuhkan pengujian koneksi yang terjadi antara web server dengan *database* serta pengujian pengambilan data yang terdapat pada *database*.

Ketika *database* dapat terkoneksi dengan web server, maka penulis dapat menggunakan data yang terdapat pada *database* melalui web server sebagai penjebaran antara aplikasi sinyal jantung dengan penyimpan data dalam hal ini *daabase* jantung.

### 4.1.1 Pengujian Koneksi

Dalam pembuatan *Web Server* perlu dilakukan koneksi sebuah *web* dengan *database* yang digunakan sebagai sumber data. Maka, dilakukan pengujian koneksi antara *web* dengan *database* yang terdapat pada PHPmyAdmin. Maka dibuatlah sebuah file untuk melakukan koneksi antara *web* dengan *database*.

The image shows a screenshot of a Sublime Text editor window titled 'C:\xampp\htdocs\jantung\connect.php - Sublime Text (UNREGISTERED)'. The editor displays PHP code for connecting to a MySQL database. The code is as follows:

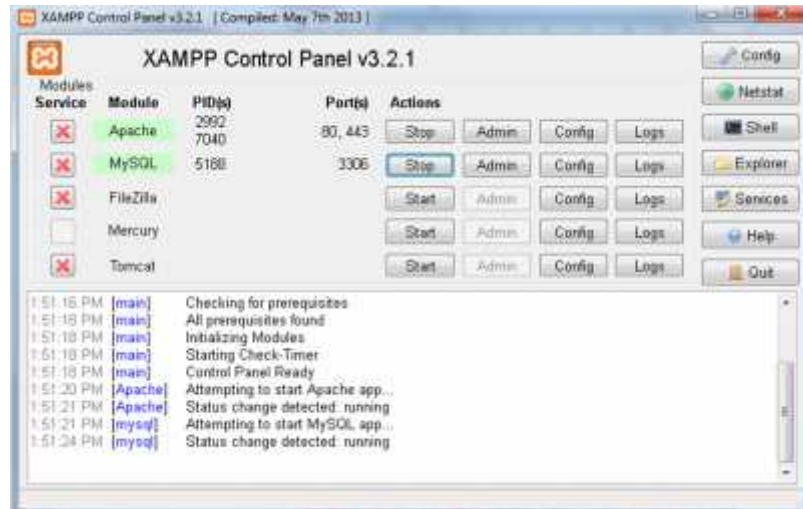
```
1 <?php
2 define('HOST','localhost');
3 define('USER','root');
4 define('PASS','');
5 define('DB','jantung_db');
6
7 $con = mysql_connect(HOST,USER,PASS,DB) or die('Unable to Connect');
8 ?>
```

The editor interface includes a menu bar with 'File', 'Edit', 'Selection', 'Find', 'View', 'Goto', 'Tools', 'Project', 'Preferences', and 'Help'. There are also several tabs open at the top, including 'views.php', 'connect.php', 'jantung.php', and 'view'.

**Gambar 4.1** Koneksi *Database* dengan *Web*

Yang dilakukan untuk melakukan koneksi *database* dengan *web* adalah dengan menyambungkan password yang terdapat pada localhost dengan database yang telah dibuat pada phpmyadmin.

Sebelum melakukan pengujian, maka yang harus dipastikan adalah modul apache dengan modul mysql pada xampp harus dalam keadaan aktif.



Gambar 4.2 XAMPP control



Gambar 4.3 Tampilan ketika koneksi berhasil



Gambar 4.4 Tampilan ketika koneksi gagal

Perbedaan antara Gambar 4.3 dengan Gambar 4.4 adalah pada Gambar 4.3 tidak ditemukan URL yang ditampilkan.

- URL Gambar 4.3 : localhost/jantung/**connect.php**
- URL Gambar 4.4 : localhost/jantung/**conect.php**

Lokasi penyimpanan **connect.php** diletakkan pada `C:\xampp\htdocs\jantung\connect.php` maka pemanggilan harus dilakukan sesuai dengan lokasi penyimpanan file `connect.php`.

#### 4.1.2 Pengujian Pengambilan Data

Data yang terdapat pada *database* diolah menjadi sebuah variabel yang nantinya akan ditampilkan kedalam sebuah aplikasi pada android. Maka dibuatlah sebuah file untuk pemisahan data pada *database*, karena android hanya memerlukan data angka pada *database* untuk ditampilkan pada android.



|   | id | nama  | value               | created_at                 |
|---|----|-------|---------------------|----------------------------|
| <input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete | 1  | edwin | edwin#0.1025390625# | 2016-07-20 17:46:24.000000 |
| <input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete | 2  | edwin | edwin#0.0927734375# | 2016-07-20 17:46:24.000000 |
| <input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete | 3  | edwin | edwin#0.1123046875# | 2016-07-20 17:46:24.000000 |
| <input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete | 4  | edwin | edwin#0.1171875#    | 2016-07-20 17:46:24.000000 |
| <input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete | 5  | edwin | edwin#0.068369375#  | 2016-07-20 17:46:24.000000 |
| <input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete | 6  | edwin | edwin#0.1123046875# | 2016-07-20 17:46:24.000000 |
| <input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete | 7  | edwin | edwin#0.1416015625# | 2016-07-20 17:46:24.000000 |
| <input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete | 8  | edwin | edwin#0.185546875#  | 2016-07-20 17:46:24.000000 |

**Gambar 4.5** Data pada *Database*

Data yang dibutuhkan untuk pembuatan grafik pada android adalah data pada kolom `value`. Nama didepan tanda pagar “#” sebagai pemilih data. Sedangkan angka atau nilai yang berada diantara pagar digunakan sebagai nilai pembuat grafik. Maka dilakukan pemisahan dan pengambilan data pada *database* dengan pembuatan program seperti ini :

```

<?php

// $char = 'A';
$char = $_GET['name'];

$sql = "SELECT * FROM datajantung WHERE value like
      '%" . $char . "%' ";

require_once('connect.php');

$r = mysqli_query($con,$sql);

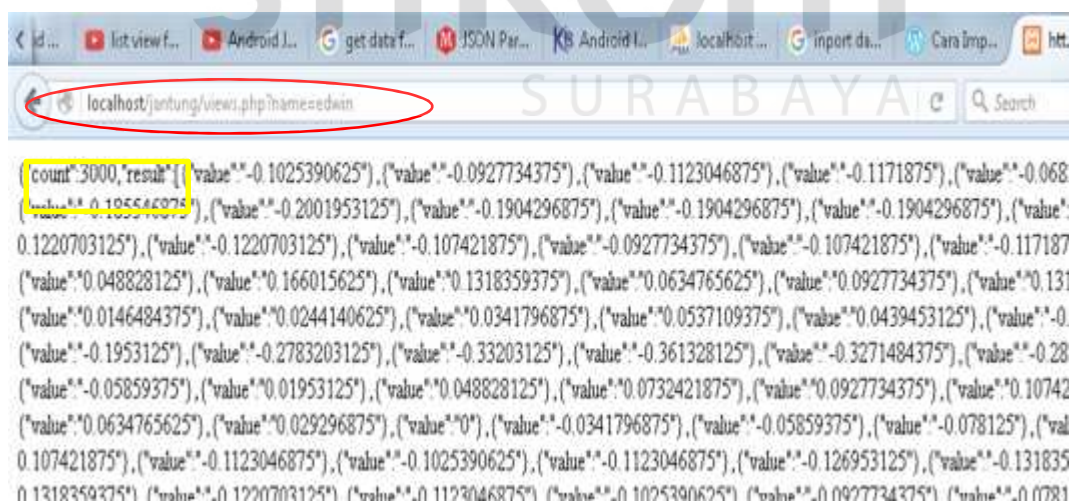
$result = array();

while($row = mysqli_fetch_array($r)){
    $value = explode("#", $row['value']);
    array_push($result,array(
        'value'=>$value[1],
    ));
}

echo json_encode(array('count' => $r->num_rows,
    'result'=>$result));
mysqli_close($con);
?>

```

Setelah dilakukan pemisahan data maka ketika dilakukan pemanggilan sesuai dengan nama pasien akan tampil data seperti pada Gambar 4.6 dibawah ini



**Gambar 4.6** Tampilan Hasil Query

Pada saat user ingin memanggil data atas nama Edwin, maka secara otomatis program akan menampilkan seluruh nilai (yang berada diantara tanda

pagar “#” pada kolom value pada *database*) beserta dengan total data dengan nama Edwin. seperti pada gambar 4.6 yang terdapat pada kotak berwarna kuning.

Berbeda halnya ketika data tidak ditemukan maka *web* akan menampilkan tampilan pada gambar 4.7 dibawah ini ;



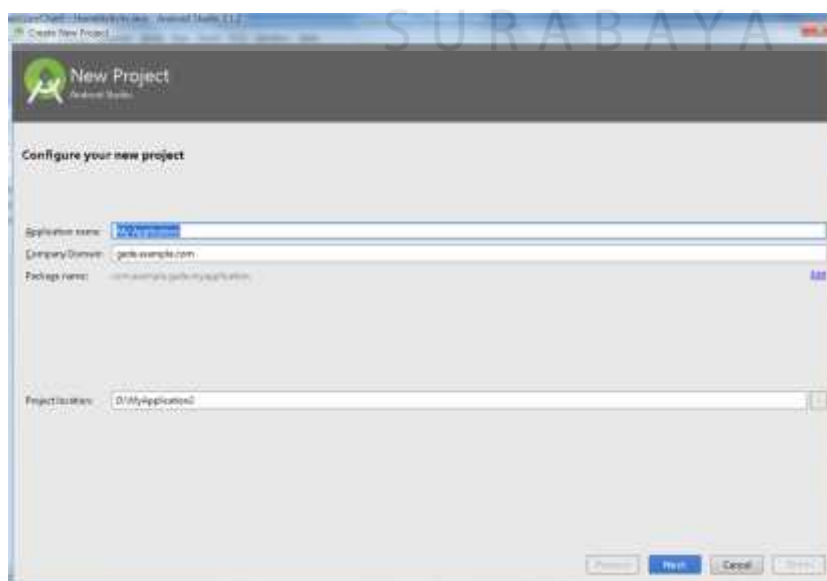
**Gambar 4.7** Tampilan Ketika Data Tidak Ditemukan

## 4.2 Pengujian Aplikasi Visualisasi Sinyal Jantung

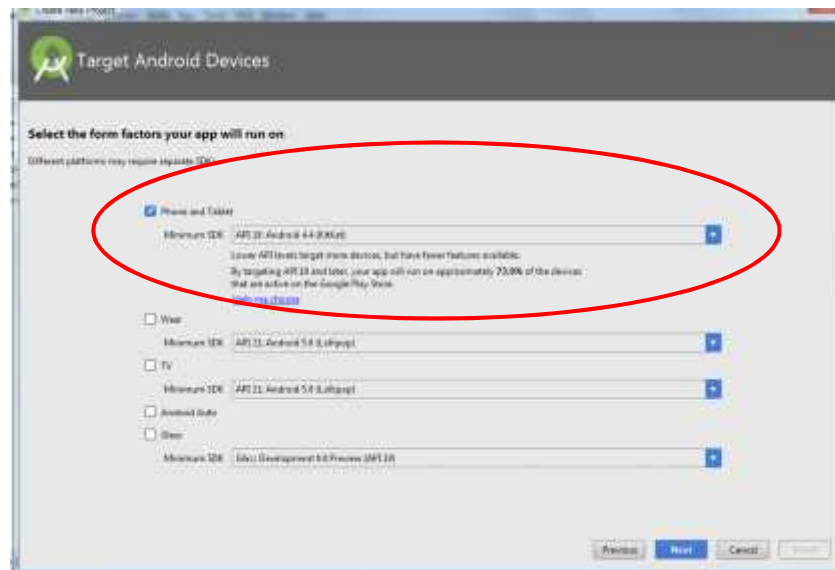
Sebelum pembuatan aplikasi pada android secara kompleks maka dilakukan terlebih dahulu beberapa pengujian untuk menguji beberapa kemampuan android.

### 4.2.1 Pembuatan Aplikasi pada *Smartphone*

Untuk pembuatan aplikasi, yang pertama kali dilakukan adalah membuat project baru.

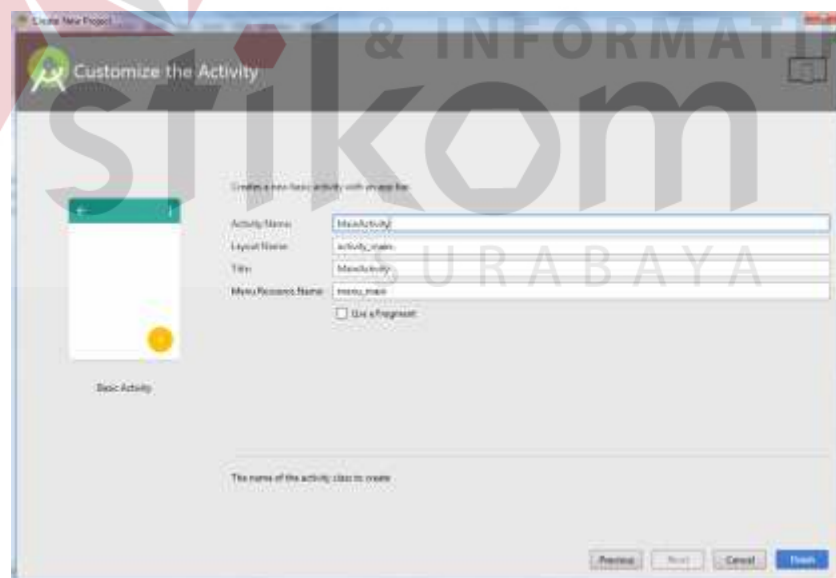


**Gambar 4.8** Pembuatan New Project



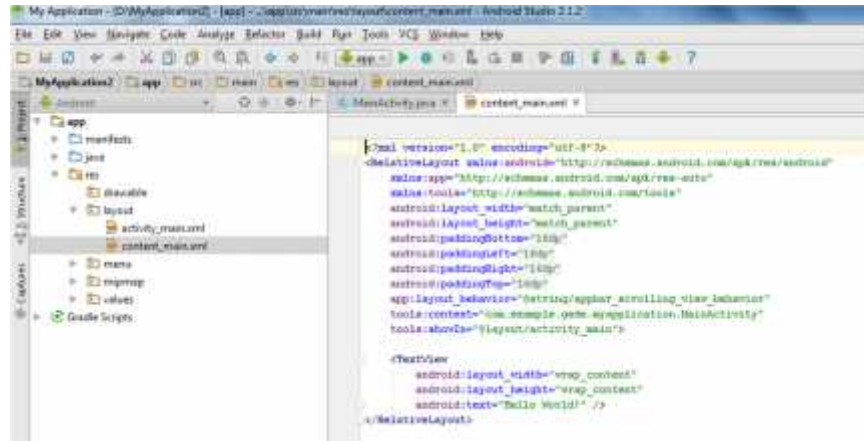
**Gambar 4.9** Pemilihan SDK minimum

Pemilihan sdk minimum sangat penting karena itu menentukan versi apa saja yang dapat diinstal oleh aplikasi ini. selanjutnya lakukan pemilihan basic project. Basic project digunakan sebagai tempat untuk melakukan pengcodangan.



**Gambar 4.10** Customize Activity

Secara otomatis android studio akan membuat sebuah project dengan seluruh propertisnya.yang diggunakan untuk membangun sebuah aplikasi pada android.



**Gambar 4.11** Tampilan Awal pada Android Studio

Sebelum program dijalankan yang perlu dipastikan adalah komputer dengan *smartphone* harus berada dalam satu jaringan. Setelah didownload maka tampilah hasil dari program diatas adalah seperti pada Gambar 4.12 dibawah ini.



**Gambar 4.12** Tampilan awalan Aplikasi Sederhana

#### 4.2.2 Pengujian Koneksi *Smartphone* dengan *Web Server*

Data jantung disimpan kedalam sebuah database pada *web server*, maka diperlukan adanya koneksi antara *web server* dengan android studio. Koneksi ini diletakkan pada file pada android studio.



```

btn_submit.setOnClickListener(new View.OnClickListener()
{
    @Override
    public void onClick(View v) {

        if(!edit_name.getText().toString().trim().equals("")){

            URL =
            "http://192.168.1.3/jantung/views.php?name=" +
            edit_name.getText();
            // Toast.makeText(getApplicationContext(),
            URL, Toast.LENGTH_LONG).show();
            accessWebService();

        } else {
            Toast.makeText(getApplicationContext(),
            "Input some text please", Toast.LENGTH_LONG).show();
        }

    }
});

```

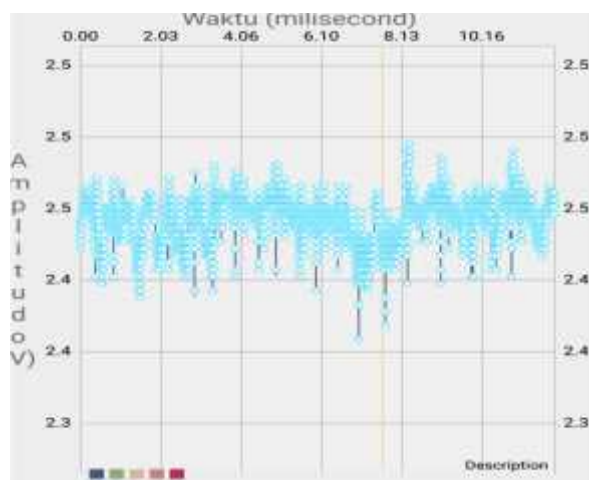
Koneksi akan terjadi ketika inputan dapat mengambil data pada android ketika data tersebut ada dan tidak menampilkan apapun ketika data tidak ada seperti Gambar 4.13 dibawah ini.

**Gambar 4.13** Koneksi android dengan *web server* berhasil

Dari gambar 4.13 tersebut diketahui bahwa koneksi berhasil dilihat dari inputan dapat menampilkan jumlah total data pada database pada *web server*.

### 4.2.3 Pengujian Tampilan Grafik Pada *Smartphone*

Pengujian grafik pada android dapat dilihat pada gambar 4.14 dibawah ini:



**Gambar 4.14** Tampilan grafik pada *Smartphone*

Pada gambar 4.14 diatas merupakan tampilan grafik pada *smartphone*.

Lingkaran – lingkaran biru pada gambar tersebut , merupakan value dari setiap data yang direpresentasikan kedalam sebuah grafik. *User* akan dapat melihat berapa nilai – nilai yang membentuk grafik tersebut ketika melakukan zoom terhadap aplikasi tersebut. Grafik dibuat dari *library* yang ditambahkan kedalam program pembuat aplikasi android, sehingga properties yang digunakan sesuai dengan properties yang terpadat pada grafik tersebut.

## 4.3 Pengujian Seluruh Program

Pada bagian ini pengujian akan dilakukan untuk semua tampilan program.

### 4.3.1 Pengujian Hasil Tampilan

Step – step aplikasi pada saat aplikasi dijalankan dapat dilihat pada gambar 4.15 sampai dengan gambar 4.17.

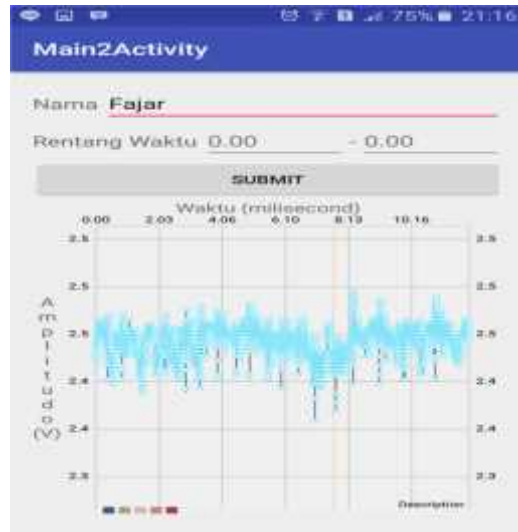


**Gambar 4.15** Tampilan awal saat aplikasi dibuka

Pada gambar 4.15 adalah tampilan awal saat aplikasi dibuka dan pada saat button masuk ditekan maka akan tampil Gambar seperti pada gambar 4.16 dibawah ini



**Gambar 4.16** Tampilan Isian program



**Gambar 4.17** Tampilan setelah data diinputkan

Pada gambar 4.17 adalah tampilan grafik setelah data diinputkan dan semua data yang ada didalam database yang sesuai dengan inputan akan dijadikan grafik

#### 4.3.2 Pengujian Waktu Penampilan Grafik

| Data       | Waktu       |             |
|------------|-------------|-------------|
|            | Lollipop    | Marshmallow |
| 3000       | 7.00 detik  | 6.36 detik  |
| 5000       | 7.80 detik  | 6.44 detik  |
| 6000       | 8.06 detik  | 7.61 detik  |
| 7000       | 9.01 detik  | 8.30 detik  |
| 9000       | 15.02 detik | 10.75 detik |
| Rata- rata | 9.58 detik  | 7.89 detik  |

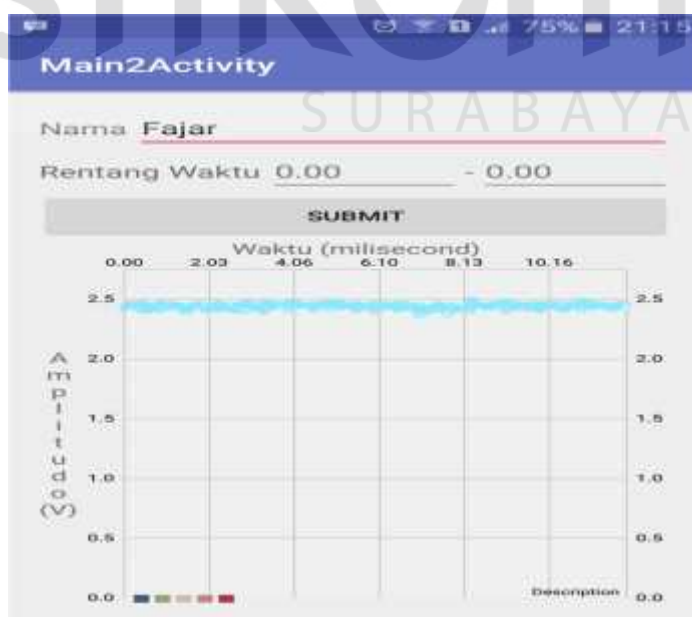
**Tabel 4.1** Hasil Pengujian Keseluruhan

Pada table 4.1 diatas, dijelaskan terdapat 5 kali pengujian *sample* data, terhadap 2 versi android pada *smartphone* yaitu lollipop dan marshmallow. Pengujian waktu dimulai pada saat button submit ditekan, karena pada saat tersebut aplikasi disambungkan kedalam *database*. Dan *database* akan menyeleksi apakah terdapat data pada *database*.

Pada table diatas dapat diketahui bahwa rata – rata kecepatan penampilan grafik pada marshmallow adalah 7.89 detik, sedangkan rata – rata keceatan penampilan grafik pada android versi lollipop adalah 9.58 detik. Semakin banyak data juga mempengaruhi kecepatan penampilan data hal ini dapat dilihat dari waktu penampilan 3000 data pada android lollipop adalah 7.00 detik sedangkan waktu penampilan 9000 data dapat tampil sempurna dalam waktu 15.2 detik. Semakin banyak data juga mempengaruhi kecepatan penampilan data hal ini dapat dilihat dari waktu penampilan 3000 data pada android Marshmallow adalah 6.36 detik sedangkan waktu penampilan 9000 data dapat tampil sempurna dalam waktu 10.75 detik

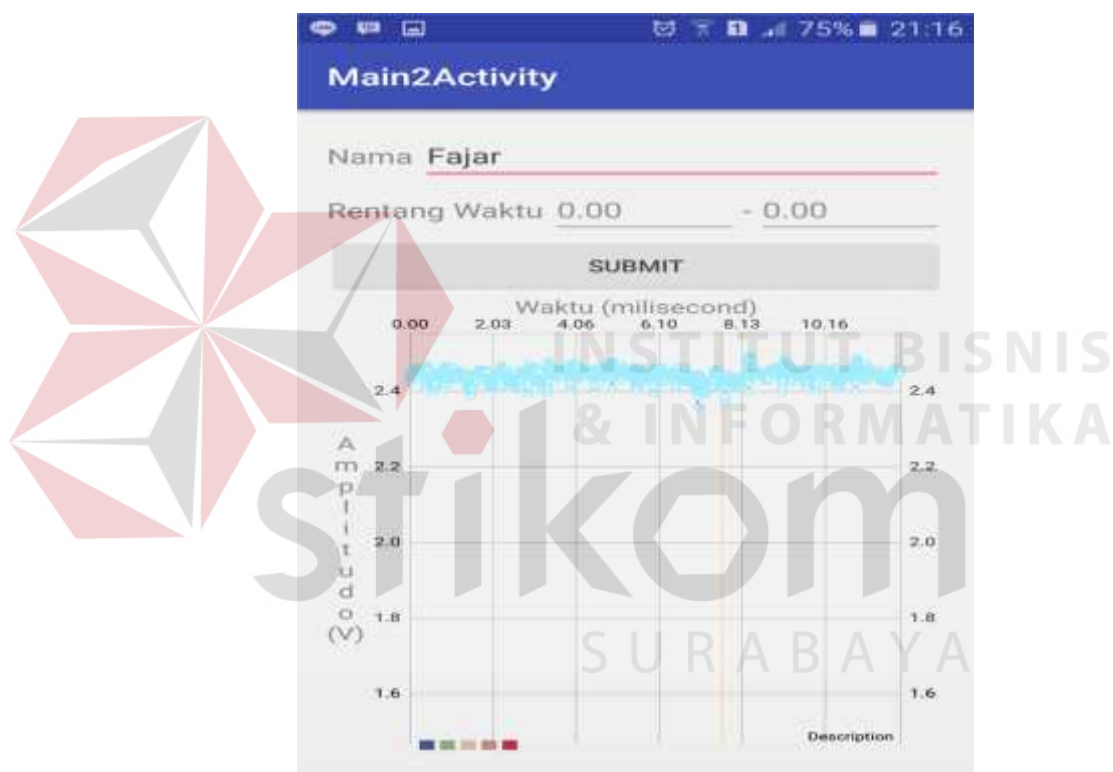
### 4.3.3 Pengujian Zoom

Pengujian zoom dilakukan karena pada aplikasi ini dibutuhkan kemampuan zooming, karena terdapat beberapa ukuran layar pada *smartphone*. Zooming dapat dilihat pada gambar 4.18 sampai dengan gambar 4.21 dibawah ini.

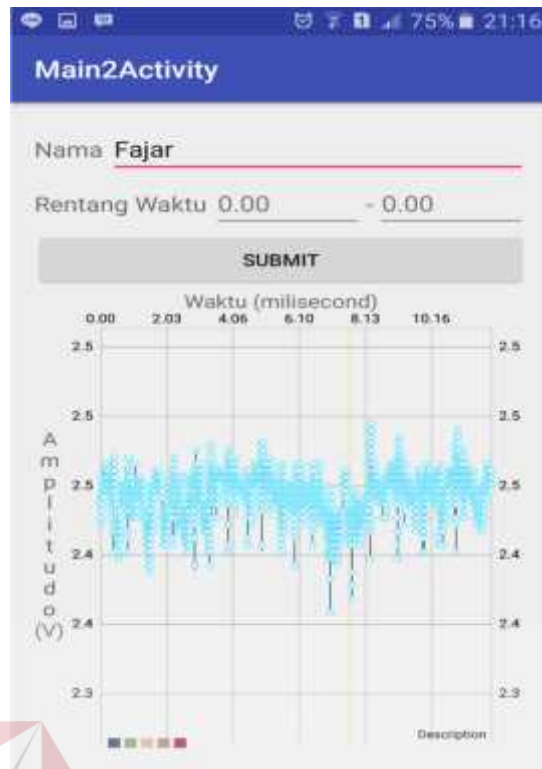


**Gambar 4.18** Grafik sebelum dilakukan zoom

Zoom hanya dapat dilakukan satu arah, seperti yang ditunjukkan pada gambar 4.19 zoom hanya dapat dilakukan secara vertical, yaitu dengan cara melebarkan layar dari bawah ke atas. Dengan jari telunjuk dan juga ibu jari, maka gambar grafik akan melebar ke atas. Hal ini dapat dilihat pada gambar 4.19 memiliki range amplitude yang lebih mendetail dibandingkan dengan pada gambar 4.18. Gambar grafik memanjang kearah atas dan kearah bawah secara bersamaan dan dengan pembesaran yang sama ke atas dan kebawah.

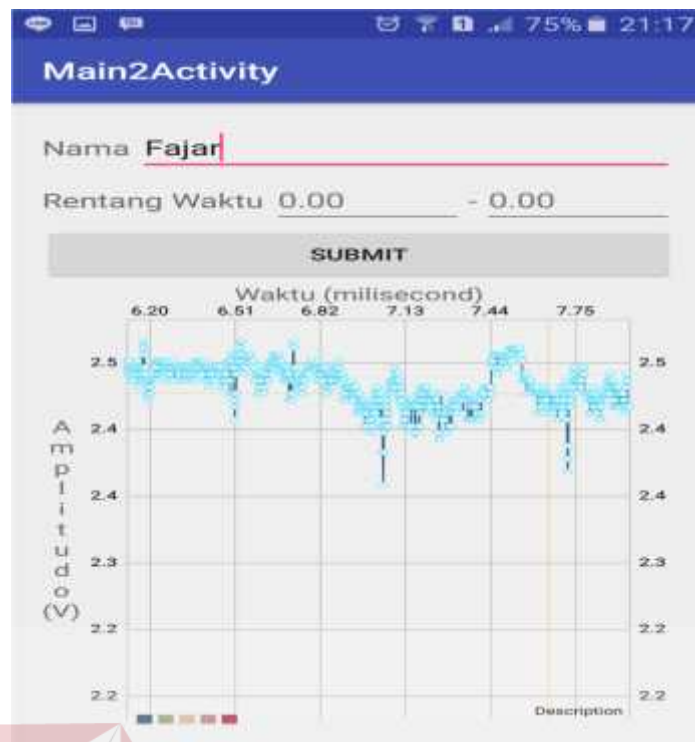


**Gambar 4.19** Grafik setelah dilakukan 1x zoom secara vertikal



**Gambar 4.20** Grafik setelah dilakukan 2x zoom secara vertical

Seperti yang telah dijelaskan pada gambar 4.19 hal yang sama juga terjadi pada gambar 4.20 diatas, gambar grafik lebih mendetail keatas dan kebawah, hanya karena data tidak dapat menampilkan secara mendetail secara angka karena kurangnya kemampuan grafik menampilkan pada layar smartphone. Tapi *user* dapat melihat secara implisit dari gambar yang ditampilkan.



**Gambar 4.21** Grafik setelah dilakukan 1x zoom secara horizontal

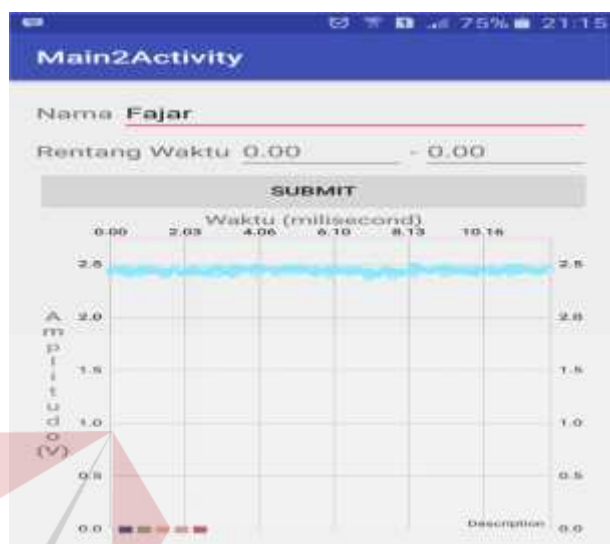
Pada gambar 4.21 adalah tampilan grafik ketika dilakukan zoom secara horizontal. Hal ini dilakukan dengan cara yang sama dengan zoom vertical, hanya saja penarikan jari telunjuk dengan ibu jari dilakukan secara horizontal (kekanan dan kekiri) dan yang terjadi adalah gambar melebar kekanan dan kekiri dengan pembesaran yang sama besar.

Dari gambar 4.18 sampai dengan gambar 4.21 dapat disimpulkan, kemampuan zoom pada aplikasi ini hanya dapat dilakukan secara vertical atau horizontal, dan tidak dapat dilakukan zoom vertical dan horizontal secara bersamaan. Dan pembesaran gambar yang terdapat pada aplikasi tidak dapat dihitung karena semakin lama *user* melakukan zoom, maka aplikasi akan terus membuat besar gambar begitu juga sebaliknya ketika *user* melakukan zoom out.



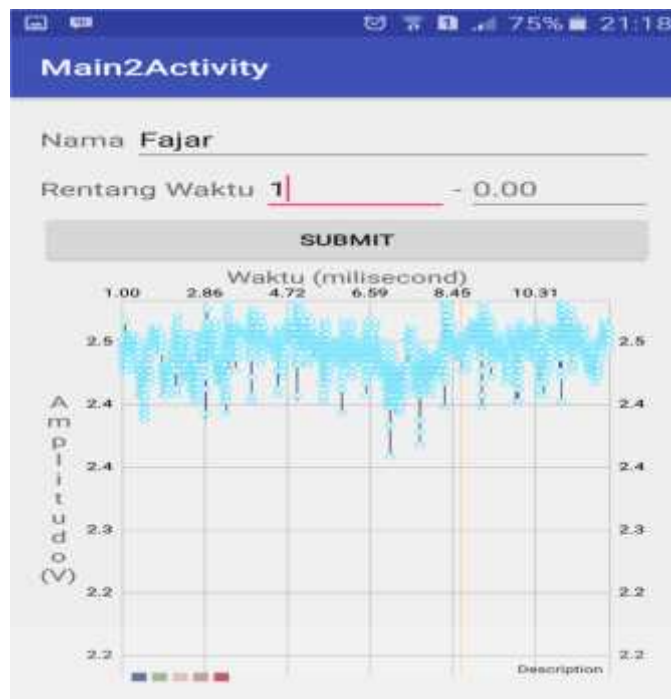
#### 4.3.4 Pengujian Filter Data

Pengujian filter data dilakukan karena dibutuhkannya gambar yang lebih detail dari grafik yang ditampilkan pada *smartphone*. Filter data dapat dilihat pada gambar 4.22 sampai dengan gambar 4.25 dibawah ini.



**Gambar 4.22** Grafik Sebelum dilakukan filter data

Filter data disini dapat digunakan dengan mengisi rentang waktu yang diinginkan oleh user. Rentang waktu dapat dilihat pada waktu yang ditampilkan pada grafik. Pada gambar 4.22 adalah gambar dimana user belum melakukan filter data rentang waktunya masih belum diberikan inputan, Inputan yang harus diisikan oleh user berdasan waktu yang ditampilkan oleh grafik. Ada dua inputan yang bisa di masukan pada rentang waktu yaitu waktu awal yang diinginkan dan waktu akhir yang user inginkan. User juga dapat mengisi keduanya secara bersamaan.



**Gambar 4.23** Inputan pada rentang waktu awal

Pada gambar 4.23 adalah tampilan grafik pada saat user menginputkan waktu awal yang diinginkan tetapi user tidak mengisikan rentang waktu akhir dan tombol submit sudah ditekan maka grafik akan berubah sesuai inputan. Jika user tidak menekan tombol submit gambar tidak akan menampilkan rentang waktu yang diinginkan user. Jika user tidak memberikan waktu akhir grafik akan menampilkan dari waktu awal yang diinginkan user sampai dengan waktu terakhir dari semua data yang ada.



**Gambar 4.24** Inputan pada rentang waktu akhir

Pada gambar 4.24 adalah tampilan grafik pada saat user menginputkan waktu akhir yang diinginkan tetapi user tidak mengisi waktu awal dan tombol submit sudah ditekan maka grafik akan berubah sesuai inputan. Jika user tidak menekan tombol submit gambar tidak akan menampilkan rentang waktu yang diinginkan user. Apabila user tidak memberikan waktu awal pada rentang waktu maka grafik yang tampil adalah dari waktu awal semua data sampai waktu akhir yang telah diinputkan oleh user.



**Gambar 4.25** Inputan pada rentang waktu awal dan waktu akhir

Pada gambar 4.25 adalah tampilan grafik pada saat user menginputkan waktu awal dan waktu akhir dan sudah ditekan tombol submit maka grafik akan berubah sesuai inputan yang telah dimasukkan oleh user.

Dari gambar 4.22 sampai dengan gambar 4.25 dapat disimpulkan, kemampuan filter data pada aplikasi ini dapat menampilkan grafik yang dibutuhkan oleh user dan sesuai dengan inputan yang diinginkan. Pada saat filter data kemampuan zoom juga dapat digunakan untuk menampilkan gambar yang lebih detail