

# PERANCANGAN APLIKASI ANDROID REKAPITULASI HASIL PEMILU SEMENTARA MENGGUNAKAN ALGORITHMMA SEQUENTIAL SERCING BERBASIS MOBILE

Fitri Latifah<sup>1</sup>, Ari Setiawan Abimanyu<sup>2</sup>,

<sup>1</sup>Komputerisasi Akuntansi, AMIK BSI Jakarta  
Jln RS Fatmawati Pondok Labu Jakarta Selatan  
[fitri.flr@bsi.ac.id](mailto:fitri.flr@bsi.ac.id)

<sup>2</sup>Teknik Informatika, STMIK Nusa Mandiri Jakarta,  
Jl. Kramat Raya No. 18, Jakarta Pusat  
[spix26@gmail.com](mailto:spix26@gmail.com)

**Abstract-** *In the conventional process of recapitulation of the ballot by collecting data from TPS to the village, then the sub-district, district to the Central Election Commission. The process is so lengthy and very time consuming. With this application input data the vote counting at polling stations in question for each candidate or candidate performed directly in a mobile polling station concerned when the counting of votes at polling process is completed. The process of data storage from the user application to the database server using JSON data exchange methods are relatively mild. In addition, the presentation of data recapitulation ballot conducted while officially takes a long time because of the lengthy process of collecting ballots also from TPS to the Central Election Commission. The design of applications that the authors have developed will make it easier to see the results in real recapitulation in text and graphics. Presentation of the recapitulation comes to the data that resides on the database server that always accumulates each input data from each polling station entrance.*

**Keywords:** *android, elections, JSON, recapitulation, electronic tabulation.*

**Intisari-** Secara konvensional proses rekapitulasi hasil suara pemilu dilakukan dengan mengumpulkan data dari TPS ke kelurahan, kemudian kecamatan, kabupaten hingga KPU Pusat. Prosesnya begitu panjang dan sangat memakan waktu. Dengan

aplikasi ini penginputan data hasil perhitungan suara di TPS yang bersangkutan untuk masing-masing calon atau kandidat dilakukan langsung secara mobile di TPS bersangkutan saat proses perhitungan suara di TPS selesai. Proses penyimpanan data dari pengguna aplikasi ke database server menggunakan metode pertukaran data JSON yang relatif ringan. Selain itu, penyajian data rekapitulasi hasil suara pemilu sementara yang dilakukan secara resmi membutuhkan waktu yang lama dikarenakan lamanya juga proses pengumpulan kertas suara dari TPS hingga KPU Pusat. Rancangan aplikasi yang penulis kembangkan akan memudahkan untuk melihat hasil rekapitulasi secara real dalam teks maupun grafik. Penyajian hasil rekapitulasi bersumber pada data yang berada pada database server yang selalu terakumulasi setiap input data dari masing-masing TPS masuk.

Kata Kunci: metode pertukaran data, KPU, TPS.

## PENDAHULUAN

Berdasarkan ketentuan UU Pemilu Legislatif dan UU Pilpres, hasil akhir yang resmi dan abash adalah perhitungan manual yang ditetapkan oleh Komisi Pemilihan Umum (KPU). Prosedur yang harus ditempuh KPU untuk menghitung seluruh suara secara manual mulai dari tingkat TPS, PPS, PPK, KPUD Kabupaten/ Kota, KPUD Provinsi, hingga KPU Pusat, hasil akhir relatif baru bias diumumkan ke publik beberapa hari kemudian setelah "Hari-H" pemungutan suara. Prosedur ini sangat

kompleks mengakibatkan lamanya informasi akan diperoleh masyarakat. Kondisi penghitungan KPU membuat masyarakat sebagai pemilih sedikit kecewa dengan proses perhitungan yang relatif lama. Kecenderungan masyarakat untuk mengetahui hasil pemilihan umum sangatlah besar sehingga kestabilan politik cenderung bergejolak.

Melihat lamanya waktu untuk hasil rekapitulasi resmi tersebut, diadakanlah *Quick Count/* Perhitungan Cepat yang dilakukan oleh lembaga survei untuk mendapatkan hasil rekapitulasementara dari pilkada yang telah dilangsungkan. Dengan masalah-masalah tersebut dapat disimpulkan bahwa kendala utama dari pelaksanaan pemilu adalah lamanya proses perhitungan dan penyajian data hasil pemilu. Berdasarkan masalah yang ada maka penulis mencoba untuk mengidentifikasi apakah aplikasi rekapitulasi hasil pemilu sementara berbasis android dapat dibangun dengan menggunakan bahasa java serta dapat diimplementasikan pada smartphone dengan sistem operasi android.

## BAHAN DAN METODE

### 1. Konsep dasar program

Program adalah kumpulan instruksi yang disusun menjadi satu kesatuan prosedur yang berurutan langkah yang disusun secara logis dan sistematis untuk menyelesaikan masalah yang diimplementasikan dengan menggunakan bahasa pemrograman, sehingga dapat dieksekusi oleh suatu program. (Jogiyanto, 2005)

### 2. Bahasa pemrograman Java

Java merupakan bahasa pemrograman berorientasi objek yang dikembangkan oleh Sun Microsystems, suatu perusahaan yang terkenal dengan Workstation UNIX high-end (Nazruddin, 2013). Java merupakan program yang sudah termasuk dalam *repository* dalam setiap versi Ubuntu.

### 3. Android

Android adalah sistem operasi yang bersifat open source. Sistem operasi ini memiliki lisensi Apache yang sangat terbuka dan bebas.

### 4. Unified Modelling Language

*Unified Modelling Language* (UML) adalah salah satu standard bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

### 5. Algoritma Sequential Searching

Dalam ilmu komputer algoritma pencarian atau algoritma sequential searching merupakan sebuah algoritma yang digunakan untuk mencari sebuah elemen data yang dilakukan secara berurutan, dengan cara melakukan perbandingan elemen satu persatu secara berurutan sampai elemen yang dicari ditemukan. Pada prinsipnya algoritma ini melakukan pengulangan atau proses *looping* mulai dari awal sampai elemen yang dicari ditemukan.

## HASIL DAN PEMBAHASAN

### Analisa dan pembahasan

Dari permasalahan yang ada dibutuhkan aplikasi mobile berbasis android yang memudahkan untuk melakukan penginputan hasil rekapitulasi suara sementara sekaligus penyajiannya dalam bentuk data maupun tabulasi.

#### a. Metode pengumpulan data

Pada penyusunan artikel ini penulis menggunakan 3 metode pengumpulan data :

##### i. Observasi

Penulis melakukan pengamatan kepada object penelitian secara langsung di lapangan.

##### ii. Wawancara

Wawancara dilakukan kepada pelaksana di lapangan untuk mendapatkan data dari permasalahan di lapangan.

##### iii. Studi pustaka

Pencarian informasi lebih lanjut dengan menggunakan referensi pendukung baik berupa literature *online* maupun *offline*

## b. Analisa penelitian dan pembahasan

### i. Identifikasi masalah

Dari masalah yang ada saat proses rekapitulasi suara pemilu sementara, maka diperlukan aplikasi berbasis android yang mendukung untuk melakukan proses rekapitulasi dan penyajian data hasil rekapitulasi pemilu sementara secara cepat.

### ii. Kebutuhan hardware dan software

1. Komputer Desktop / Laptop  
Spesifikasi :
  - a. Prosesor minimum Core2Duo
  - b. RAM 2GB
  - c. VGA Card 512 MB
  - d. LAN Card 100mbps
  - e. Wifi / Internet
2. Ponsel bersistem operasi Android  
Spesifikasi :
  - a. Processor Dual Core
  - b. RAM 768 MB
  - c. Wifi / akses internet
  - d. Minimum versi Android Froyo
3. Software :
  - a. Eclipse 3.6.2 (Helios ) atau lebih tinggi
  - b. Eclipse JDT plugin
  - c. Java Development kit (JDK) versi 6
  - d. Android Development Tools plugin
  - e. Apache Web Server
  - f. MySQL Database
  - g. SQLite Database

### iii. Algoritma pada kasus

Algoritma sekuensial adalah suatu algoritma dimana setiap baris program akan dikerjakan secara berurutan dari atas ke bawah sesuai dengan urutan penulisannya.

Algoritma yang akan digunakan penulis dalam aplikasi ini adalah algoritma *sequential searching*. Algoritma yang akan dijalankan aplikasi adalah sebagai berikut :

#### a. Algoritma sekuensial

1. Melakukan deklarasi atau inialisasi variabel

- a. VAR U = User String
- b. VAR C = Calon String
- c. VAR S = Suara Integer
- d. VAR P = Pesan Kesalahan
2. Memasukan nilai data
  - a. Masukkan nilai U
  - b. Masukkan nilai C
  - c. Masukkan nilai S
3. Melakukan proses penginputan  
IF ( U = Kosong ) THEN  
P = "ID User tidak boleh kosong";  
ELSE IF ( C = Kosong ) THEN  
P = "Data Calon tidak boleh kosong";  
ELSE IF ( S = Kosong ) THEN  
P = "Jumlah suara tidak boleh kosong";  
ELSE IF ( U = Ada && C = Ada && S = Ada )  
new DoInput().execute();  
P = "Berhasil Input Suara";  
ELSE  
P = "Silahkan isi data terlebih dahulu";
4. Menampilkan keterangan proses input  
ECHO P;

#### b. Algoritma sekuensial search

1. Melakukan deklarasi
  - a. VAR SQL = QUERY
  - b. VAR I = INTEGER (PENCATAT INDEKS LARIK/ARRAY)
  - c. VAR D = ARRAY();
  - d. VAR Q = PROSES QUERY
  - e. VAR DATA = VARIABEL ESCAPE ARRAY
  - f. VAR KETEMU = BOOLEAN (OUTPUT)
2. Memasukan nilai ke dalam variabel
  - a. SQL = SELECT vote.id\_calon, SUM(vote.jum\_suara), calon.nama\_calon FROM vote INNER JOIN calon ON vote.id\_calon = calon.id\_calon GROUP BY id\_calon";
  - b. I = 1;
3. Melakukan proses pencarian  
IF ( Q = TRUE ) {  
WHILE ( FETCH ( Q ) ) {  
D [ ] ;  
}  
}  
  
WHILE ( I < N ) AND ( D[ I ] < X ) DO  
I = I + 1

```

END WHILE

IF D [ I ] = X THEN {
    KETEMU = TRUE;
}

ELSE {
    D [ ] = 0;
    KETEMU = FALSE;
}

D [ ];
}

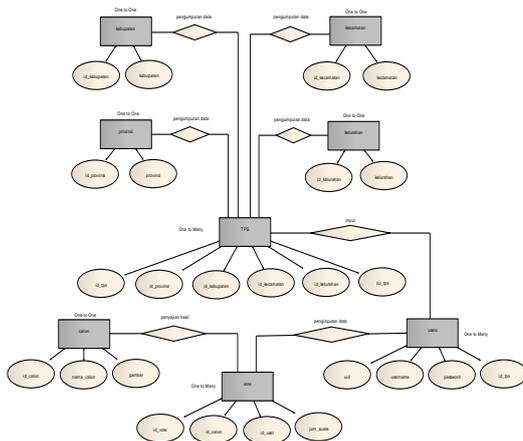
RETURN
    
```

```

4. Menampilkan hasil array hasil
   pencarian
   FOREACH ( D AS DATA ) {
       ECHO DATA[2];
       ECHO DATA[1];
   }
    
```

Pada algoritma tersebut proses akan mencari variable yang di masukkan sampai dengan variable yang dicari ditemukan. Apabila dalam proses pencarian variable tidak ditemukan maka sistem akan mencetak "data tidak ditemukan".

**iv. Rancangan database**

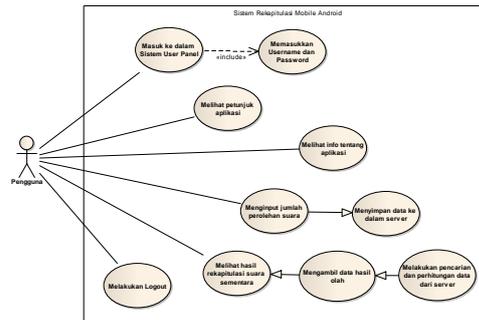


Sumber: Pengolahan Sendiri (2016)

**Gambar 1. Entity relations diagram**

Database terdiri dari 8 tabel yaitu tabel provinsi, kabupaten, kecamatan, kelurahan, tps, users, calon, dan vote.

**v. Usecase diagram**

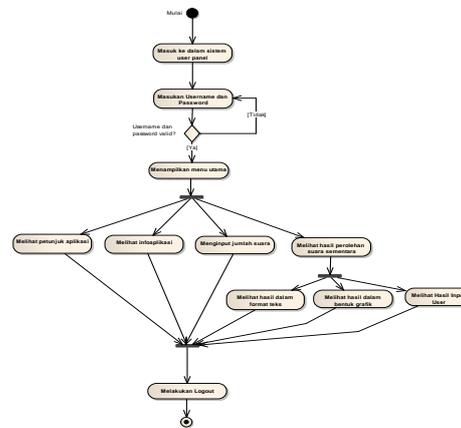


Sumber: Pengolahan Sendiri (2016)

**Gambar 2. Use Case Diagram**

User dapat memilih aktifitas apa yang akan dipilih mulai dari login, memilih melihat petunjuk dan info aplikasi, melakukan input suara, melihat hasil rekapitulasi sementara dalam bentuk teks maupun grafik, dan melakukan logout dari sistem.

**vi. Activity Diagram**

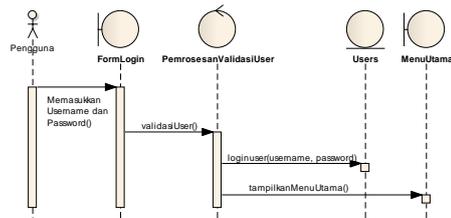


Sumber: Pengolahan Sendiri (2016)

**Gambar 3. Activity Diagram**

Activity diagram menggambarkan diagram aliran kerja pada aplikasi. User diharuskan untuk melakukan login terlebih dahulu. Setelah login user dapat memilih menu melihat petunjuk dan info aplikasi, melakukan input suara, dan melihat hasil rekapitulasi dalam bentuk teks maupun grafik.

### vii. Sequence diagram login

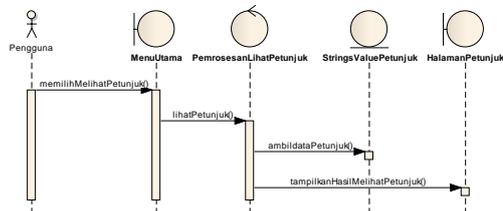


Sumber: Pengolahan Sendiri (2016)

**Gambar 4. Sequence Diagram Login**

*Sequence diagram login* menggambarkan tahapan yang diperlukan oleh *user* untuk bisa melakukan *login*. Untuk dapat melakukan *login*, *user* diharuskan memasukkan *username* dan *password*.

### viii. Sequence diagram melihat petunjuk aplikasi

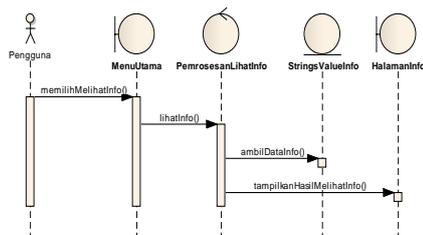


Sumber: Pengolahan Sendiri (2016)

**Gambar 5. Sequence Diagram Melihat Petunjuk Aplikasi**

*Sequence diagram* melihat petunjuk aplikasi merupakan tahapan *user* melihat petunjuk penggunaan aplikasi atau kegiatan apa yang harus dilakukan *user* sehingga penggunaan aplikasi sesuai dengan yang diharapkan.

### ix. Sequence diagram melihat info aplikasi

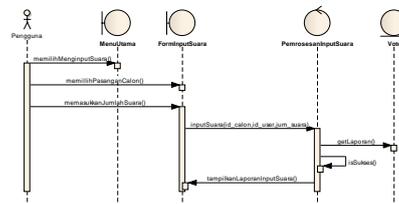


Sumber: Pengolahan Sendiri (2016)

**Gambar 6. Sequence Diagram Melihat Info Aplikasi**

*Sequence diagram* melihat info aplikasi merupakan tahapan *user* dapat melihat daftar info aplikasi.

### x. Sequence diagram input suara

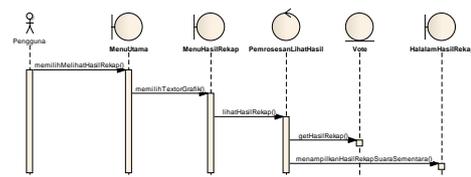


Sumber: Pengolahan Sendiri (2016)

**Gambar 7. Sequence Diagram Input Suara**

*Sequence Diagram* input suara merupakan tahapan *user* dapat melakukan input hasil suara pemilu sementara yang akan dikirimkan dan disimpan ke database server.

### xi. Sequence Diagram melihat hasil rekapitulasi

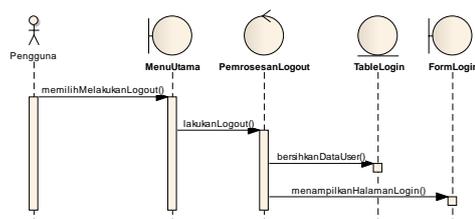


Sumber: Pengolahan Sendiri (2016)

**Gambar 8. Sequence Diagram Melihat Hasil Rekapitulasi**

*Sequence diagram* melihat hasil rekapitulasi merupakan tahapan *user* dapat melihat hasil rekapitulasi hasil perhitungan suara pemilu sementara dalam bentuk teks maupun dalam bentuk grafik.

### xii. Sequence Diagram Logout

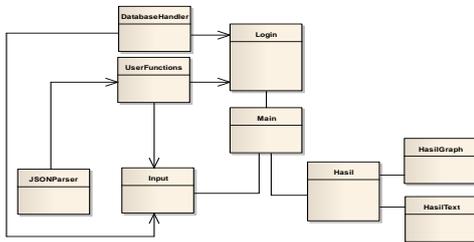


Sumber: Pengolahan Sendiri (2016)

**Gambar 9. Sequence Diagram Logout**

Sequence Diagram Logout merupakan tahapan untuk keluar dari sistem aplikasi rekapitulasi suara pemilu sementara.

**xiii. Class diagram**



Sumber: Pengolahan Sendiri (2016)

**Gambar 10. Class Diagram**

Class diagram menggambarkan alur data yang dilalui oleh aplikasi dan sekaligus pengelompokan fungsi-fungsi program.

**xiv. Deployment diagram**



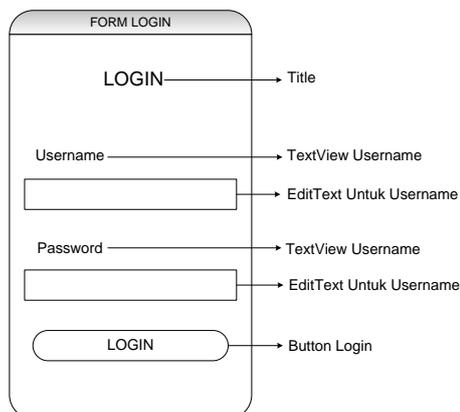
Sumber: Pengolahan Sendiri (2016)

**Gambar 11. Deployment Diagram**

Deployment diagram merupakan hubungan aplikasi yang terinstal di smartphone dengan server.

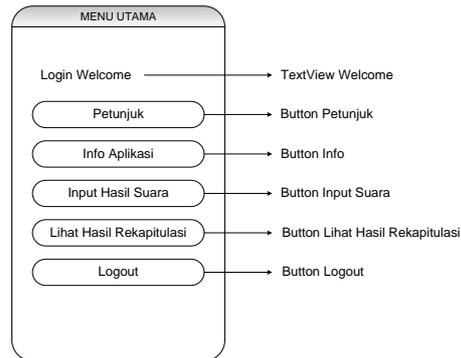
**c. User interface**

**i. Halaman login**



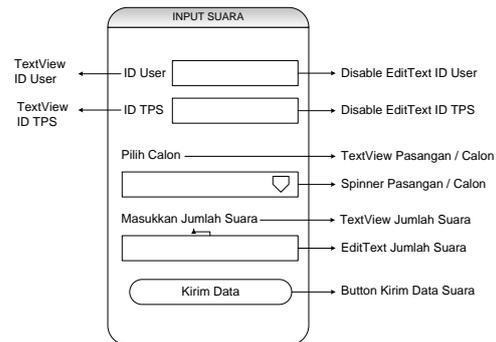
**Gambar 12. Halaman Login**

**ii. Menu utama**



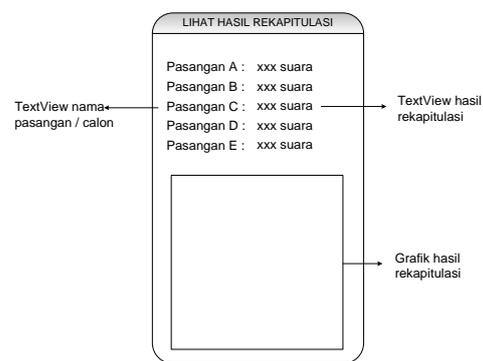
**Gambar 13. Menu Utama**

**iii. Input suara**



**Gambar 14. Input Suara**

**iv. Hasil rekapitulasi**

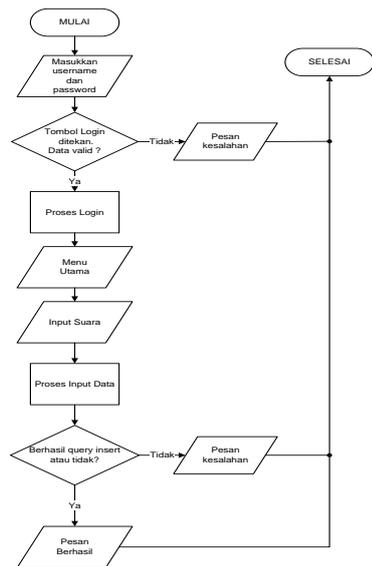


**Gambar 15. Hasil Rekapitulasi**

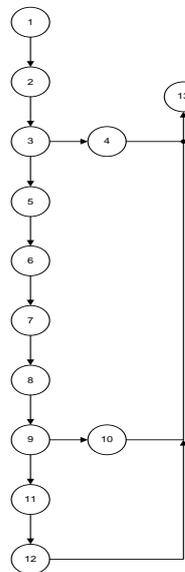
**6. Testing dan Implementasi**

**a. White box testing**

Hasil pengujian white box testing aplikasi sebagai berikut:



Gambar 16 . Flowchat *white box testing*



Gambar 17 . Diagram alir *white box testing*

Kompleksitas siklomatis dari diagram alir *white box* dapat diperoleh dengan perhitungan:

$$V(G) = E - N + 2$$

Dimana:

E = Jumlah *Edge* yang ditentukan gambar panah

N = Jumlah simpul grafik alir ditentukan dengan gambar lingkaran

V(G) = kompleksitas siklomatis

Sehingga didapat:

Nilai E = 14

Nilai N = 13

Sehingga nilai V(G)

$$V(G) = E - N + 2$$

$$V(G) = 14 - 13 + 2$$

$$V(G) = 3$$

V(G) < 10 berarti memenuhi syarat kekompleksitas siklomatisnya.

Baris set yang dihasilkan dari jalur *independent* adalah sebagai berikut:

- 1-2-3-5-6-7-8-9-11-12-13
- 1-2-3-5-6-7-8-9-10-13
- 1-2-3-4-13

Dapat disimpulkan bahwa ketika aplikasi dijalankan dapat diketahui bahwa set baris yang dihasilkan adalah 1-2-3-5-6-7-8-9-11-12-13-1-2-3-5-6-7-8-9-10-13-1-2-3-4-13, dengan demikian pada saat aplikasi jalan terlihat simpul dieksekusi satu kali.

### b. Black box testing

Tabel 1 . Hasil *black box testing*

No	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian
1	Mengosongkan kolom Username kemudian tekan tombol Login	Form Login	Sistem akan menol ak login	Valid
2	Mengosongkan kolom Password kemudian tekan tombol Login	Form Login	Sistem akan menol ak login	Valid
3	Mengosongkan kolom Username dan Password kemudian tekan tombol Login	Form Login	Sistem akan menol ak login	Valid
4	Mengisi kolom username dan password yang tidak terdapat	Form Login	Sistem akan menol ak login	Valid

	dalam database			
5.	Mengisi kolom username dan password yang terdapat dalam database	Form Login	Sistem akan memb user pada halaman menu utama	Valid
6.	Mengosongkan kolom jumlah suara	Form Input Data Perolehan Suara	Sistem akan menolak menyimpan data ke server.	Valid
7.	Menginput data pada kolom jumlah suara	Form Input Data Perolehan Suara	Sistem akan menyimpan data ke server	Valid

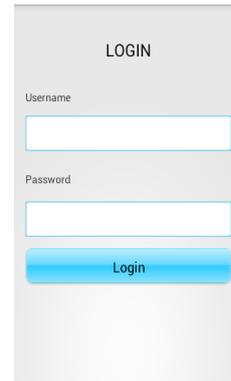
**c. Implementasi**

**i. Tampilan Splash Screen**



**Gambar 18. Splash Screen**

**ii. Tampilan form login**



**Gambar 19. Form Login**

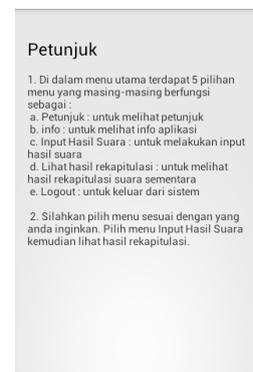
Daftar *user* merupakan menu untuk menambahkan *user* apabila *user* belum memiliki akses masuk berupa *username* dan *password*

**iii. Tampilan Menu utama**



**Gambar 20. Menu Utama**

**iv. Tampilan Petunjuk Aplikasi**



**Gambar 21. Petunjuk Aplikasi**

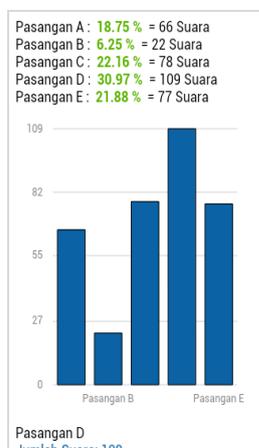
## v. Tampilan Info Aplikasi



Gambar 22. Info Aplikasi  
vi. Tampilan input data suara

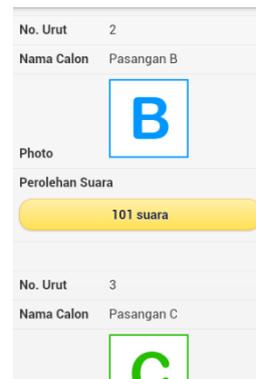
Gambar 23. Input Data Suara

## vii. Tampilan Hasil Grafik Bar



Gambar 24. Hasil Grafik Bar

## viii. Tampilan Hasil Teks



Gambar 25. Hasil Teks

## KESIMPULAN

Kesimpulan yang dapat diambil setelah merancang dan membuat serta melakukan pengujian aplikasi mobile rekapitulasi hasil suara pemilu atau real quick count berbasis android adalah Penggunaan format pertukaran data JSON merupakan pilihan yang tepat dalam membangun aplikasi android *client server* yang terpusat dalam database server karena transfer data lebih ringan sehingga penyajian hasil rekapitulasi suara sementara lebih cepat. Proses *query* untuk menyimpan data atau proses penginputan data ke database server sangat tergantung pada kecepatan paket data internet jaringan operator seluler. Penyajian data aplikasi ini akan sangat baik jika bekerja pada sinyal yang 3G. Aplikasi Rekapitulasi Mobile / Real Quick Count berbasis android ini dapat berjalan dengan baik pada versi Android 4.0 Ice Cream Sandwich. Menggunakan bahasa pemrograman OOP baik dari sisi user dengan menggunakan pemrograman Java atau dari sisi server dengan menggunakan pemrograman PHP sangatlah membantu dalam proses pembuatan program karena didukung oleh fitur-fitur yang ada pada pemrograman berbasis objek Penyajian data berupa grafik bar maupun teks adalah hasil rekapitulasi suara sementara dan bukanlah hasil akhir dari perhitungan suara yang sah. Karena hasil perhitungan suara yang sah adalah hasil perhitungan kertas suara secara manual oleh KPU.

Saran dan masukan setelah perancangan, pembuatan dan pengujian aplikasi ini agar dapat dikembangkan lebih baik dan sesuai kebutuhan di lapangan adalah: Dikarenakan tidak hanya faktor kebenaran alur data dan kesempurnaan sistem aplikasi saja, sebuah aplikasi juga dituntut agar dibuat semudah mungkin menurut cara pandang pengguna (*user friendly*) dan tidak membuat pengguna kesulitan dalam menggunakan aplikasi tersebut atau langsung mengetahui secara sekilas bagaimana menggunakan aplikasi ini. Maka dari itu perancangan antar muka / *UI (User Interface)* dan juga *UX (User Experience)* harus menjadi perhatian lebih selain kebenaran algoritma. Pembangunan server database perlu diperhatikan mengingat tingginya *load* proses ketika proses input data dari tiap TPS secara serempak. Maka dari itu spesifikasi server yang tinggi dan handal sangat dibutuhkan agar sistem ini dapat bekerja dengan baik tanpa adanya *hang* sistem. Diperlukan jaringan *private* selular agar transfer data lebih cepat dan meminimalisir pencurian data saat data dikirim ke database server. Penyajian data hasil rekapitulasi suara sementara yang telah dapat ditampilkan dalam bentuk teks maupun grafik bar, akan lebih baik jika ditambahkan filter-filter berupa wilayah seperti kabupaten, kecamatan dan kelurahan. Selain itu hasil pemilihan juga dapat ditampilkan berdasarkan penyebaran pemilih dan letak geografis wilayah.

#### REFERENSI

- Dharma, Akhmad. 2013. Kolaborasi Dahsyat ANDROID dengan PHP dan MYSQL. Jakarta: Lokomedia
- Irawan.2012. Membuat Aplikasi Android Untuk Orang Awam. Palembang.Maxikom
- Komputer, Wahana. 2013. Android Programming with Eclipse. Yogyakarta: Andi Offset
- Jogiyanto. 2005. Analisis & Desain. Yogyakarta : Andi Offset.
- Rachmat, Antonius. 2010. Algoritma dan Pemrograman dengan Bahasa C. Yogyakarta: Andi Offset.
- Safaat, Nazruddin. 2012. Android Pemrograman Aplikasi Mobile Smartphone Dan Tablet PC Berbasis Android. Bandung: Informatika.
- Safaat, Nazruddin. 2013. Aplikasi berbasis Android. Bandung: Informatika.
- Sakur, Stendy. 2010. PHP 5 Pemrograman Berorientasi Objek Konsep & Implementasi. Yogyakarta: Andi Offset.
- Sibero, Alexander F.K. 2013. Web Programming Power Pack. Yogyakarta:MediaKom.
- Sjukani, Moh. 2007. Struktur Data (Algoritma &Struktur Data 2) dengan C, C++ Edisi 3. Jakarta : Mitra Wacana Media.

#### Biodata Penulis

1. **Fitri Lataifah** adalah dosen tetap pada AMIK BSI Jakarta di program studi Komputerisasi Akuntansi, dan juga sebagai pengajar di Prodi Teknik Informatika STMIK Nusamandiri dengan matakuliah stuktur data serta teknik pemrograman
2. **Ari Setiawan Abimanyu** adalah mahasiswa program studi teknik informatika STMIK Nusamandiri jakarta