BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1. Implementasi

Implementasi *prototype* Aplikasi *E-Voting* ini dilakukan menggunakan bahasa pemrograman Java dengan Eclipse sebagai IDE(*Intregrated Development Environment*)-nya, dan juga untuk penyimpanan data nya menggunakan MySQL pada sisi *server* dan SQLite pada sisi *client*. Bahasa pemograman Java tersebut dapat dijalankan pada berbagai platform sistem operasi dan perangkat keras, tetapi implementasi dan pengujian sepenuhnya hanya dilakukan pada *smartphone* dengan Sistem Operasi Android versi 2.2.

5.1.1. Implementasi Perangkat Lunak

Untuk implementasi perangkat lunak ini adalah *handphone* atau telepon seluler yang memiliki sistem operasi android versi 2.2. ke atas dan dapat terhubung dengan *internet*. Sedangkan untuk sisi *server*, dibutuhkan beberapa perangkat lunak yang didantaranya adalah

1. *Apache*

Penulis menggunakan apache web server untuk mengolah kode PHP atau HTML.

2. *Database Server*

Database server yang digunakan oleh penulis adalah MySQL untuk menyimpan sumber data aplikasi.

3. *Netbeans*

Untuk membuat script php, yang berfungsi sebagai penghubung interaksi antara aplikasi *e-voting* pada *smartphone* android dengan basis data.

4. *Xampp*

Xampp digunakan untuk memudahkan penggunaan *apache* dan *mysql* yang dijadikan satu paket. Serta untuk memudahkan pengelolaan *database* dengan adanya *PHPMyAdmin*.

5. *Mozilla FireFox*

Mozilla FireFox, sebagai media untuk menjalankan *PHPMyAdmin* yang nantinya akan digunakan untuk mengolah database.

5.1.2. Implementasi Perangkat Keras

Untuk dapat menjalankan aplikasi yang dirancang maka dibutuhkan suatu perangkat keras sebagai penunjangnya yang dalam hal ini adalah perangkat keras yang dimaksud adalah *smartphones* dengan sistem operasi android. Adapun perangkat kebutuhan perangkat keras yang dibutuhkan tersebut adalah sebagai berikut :

1. User

a. Processor 1.4 GHz Scorpion processor, Adreno 205 GPU.
b. Network 2G : GSM 850 / 900 / 1800 / 1900, 3G: HSDPA
c. Operating System: Android 2.2 Froyo or Higher.
d. Memory: 512 MB RAM, 2 GB ROM

2. Server

a. Menggunakan p*rocessor* dengan *Front Side Bus* (FSB) minimal 800 Mhz

b. Menggunakan RAM minimal 512 MB.

c. Tersedianya *Hard Drive Space* untuk media penyimpanan minimal 50 MB diluar basis data.

d. *Mouse*, *Keyboard*, dan *Monitor* sebagai peralatan masukan dan keluaran (I/O).

5.1.3. Implementasi Basis Data

Implementasi basis data diambil berdasarkan kebutuhan data untuk aplikasi, secara fisik, implementasi basis data diimplementasikan menggunakan perangkat lunak MySQL *server*, Tabel-tabel berikut menggambarkan struktur tabel yang diimplementasikan pada basis data.

1. Tabel Fakultas

```
CREATE TABLE IF NOT EXISTS `fakultas` (
  `kode_fakultas` int(1) NOT NULL,
  `nama_fakultas` varchar(50) NOT NULL,
  `votedscore` int(10) NOT NULL DEFAULT '0',
  PRIMARY KEY (`kode_fakultas`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

2. Tabel Jurusan

```
CREATE TABLE IF NOT EXISTS `jurusan` (
  `kode_jurusan` varchar(2) NOT NULL,
  `nama_jurusan` varchar(50) NOT NULL,
  `jenjang` varchar(2) NOT NULL,
  `votedscore` int(10) NOT NULL DEFAULT '0',
  `kode_fakultas` int(1) NOT NULL,
  PRIMARY KEY (`kode_jurusan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3. Tabel Mahasiswa

```
CREATE TABLE IF NOT EXISTS `mahasiswa` (
  `nim` int(8) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `password` varchar(30) NOT NULL,
  `isvoted` int(1) NOT NULL DEFAULT '0',
  `status` varchar(8) NOT NULL DEFAULT 'offline',
  `cdate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `kode_jurusan` varchar(2) NOT NULL,
  `kode_fakultas` int(1) NOT NULL,
  PRIMARY KEY (`nim`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4. Tabel Kandidat Presiden dan Wakil Presiden Mahasiswa

```
CREATE TABLE IF NOT EXISTS `kandidat_presma` (
  `id_kandidat` int(3) NOT NULL AUTO_INCREMENT,
  `picture_presma` varchar(100) NOT NULL DEFAULT '0',
  `picture_wapres` varchar(100) NOT NULL DEFAULT '0',
  `score` int(11) NOT NULL,
  `lastvoted` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `nim_presma` int(8) NOT NULL,
  `nim_wapres` int(8) NOT NULL,
  PRIMARY KEY (`id_kandidat`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

5. Tabel Misi Kandidat

```
CREATE TABLE IF NOT EXISTS `misi_kandidat` (
  `misi` varchar(100) NOT NULL,
  `id_kandidat` int(3) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

6. Tabel Visi Kandidat

```
CREATE TABLE IF NOT EXISTS `visi_kandidat` (
  `visi` varchar(100) NOT NULL,
  `id_kandidat` int(3) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

7. Tabel Program Kerja Kandidat

```
CREATE TABLE IF NOT EXISTS `programkerja_kandidat` (
  `program_kerja` varchar(100) NOT NULL,
  `id_kandidat` int(3) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

8. Tabel Pengalaman Organisasi Kandidat

```
CREATE TABLE IF NOT EXISTS `pengalaman_organisasi_kandidat` (
  `organisasi` varchar(150) NOT NULL,
  `tahun_aktif` year(4) NOT NULL,
  `tahun_keluar` year(4) NOT NULL,
  `jabatan` varchar(150) NOT NULL,
  `nim` int(8) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

9. Tabel Prestasi Kandidat

```
CREATE TABLE IF NOT EXISTS `prestasi_kandidat` (
  `prestasi` varchar(150) NOT NULL,
  `tahun` year(4) NOT NULL,
  `nim` int(8) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

10. Tabel Broadcast

```
CREATE TABLE IF NOT EXISTS `tbl_broadcast` (
  `dari` varchar(8) NOT NULL,
  `broadcast_msg` varchar(200) NOT NULL,
  `timestamp` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

11. Tabel Message

```
CREATE TABLE IF NOT EXISTS `tbl_msg` (
  `dari` varchar(8) NOT NULL,
  `ke` varchar(8) NOT NULL,
  `msg` varchar(200) NOT NULL,
  `timestamp` timestamp NULL DEFAULT CURRENT_TIMESTAMP,
  `status` int(1) NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

12. Tabel Perolehan Suara Fakultas

```
CREATE TABLE IF NOT EXISTS `perolehan_suara_fakultas` (
  `score` int(10) NOT NULL DEFAULT '0',
  `kode_fakultas` int(1) NOT NULL,
  `id_kandidat` int(3) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

13. Tabel Perolehan Suara Jurusan

```
CREATE TABLE IF NOT EXISTS `perolehan_suara_jurusan` (
  `score` int(10) NOT NULL DEFAULT '0',
  `kode_jurusan` varchar(2) NOT NULL,
  `id_kandidat` int(3) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

14. Tabel Sosialisasi PEMIRA

```
CREATE TABLE IF NOT EXISTS `sosiliasasi_pemira` (
  `id_pemira` int(1) NOT NULL,
  `tgl_awal` varchar(10) NOT NULL COMMENT 'year-month-date',
  `tgl_akhir` varchar(10) NOT NULL COMMENT 'year-month-date',
  `ket` varchar(25) NOT NULL,
  PRIMARY KEY (`id_pemira`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

15. Tabel Administrator

```
CREATE TABLE IF NOT EXISTS `admistrator` (
  `username` varchar(20) NOT NULL,
  `password` varchar(20) NOT NULL,
  `nim_panitia` int(8) NOT NULL,
  PRIMARY KEY (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.4. Implementasi Antar Muka

Implementasi antarmuka bertujuan untuk menjelaskan bentuk dari implementasi antarmuka yang sudah dirancang sebelumnya, adapun tampilan antarmuka dari aplikasi *e-voting* ini adalah sebagai berikut.

5.1.5. Implementasi Instalasi Program

Agar aplikasi *e-voting* dapat berjalan maka perlu ada instalasi program baik di sisi *server* dan sisi perengkat *smartphone user*, diantaranya yaitu :

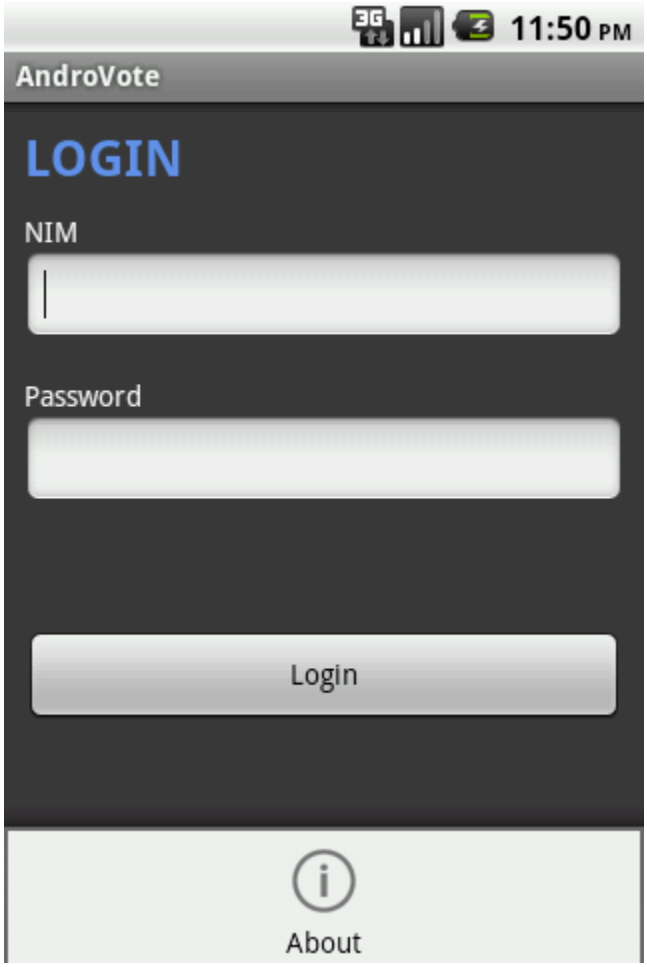
1. *Server*
   a. Meng*install* XAMPP untuk menjalankan *server* apache dan *database* MySQL
   b. Menginstall *Mozilla Firefox* atau aplikasi *browser* lainnya untuk mengakses *PHPMyAdmin* yang nantinya akan digunakan untuk mengolah *database*.
2. *User,* hanya menginstall evotingbem.apk pada perangkat android *smartphone* nya

5.1.6. Penggunaan Program

Penggunaan aplikasi *e-voting* ini, akan dijelaskan dalam beberapa tahapan di bawah ini.

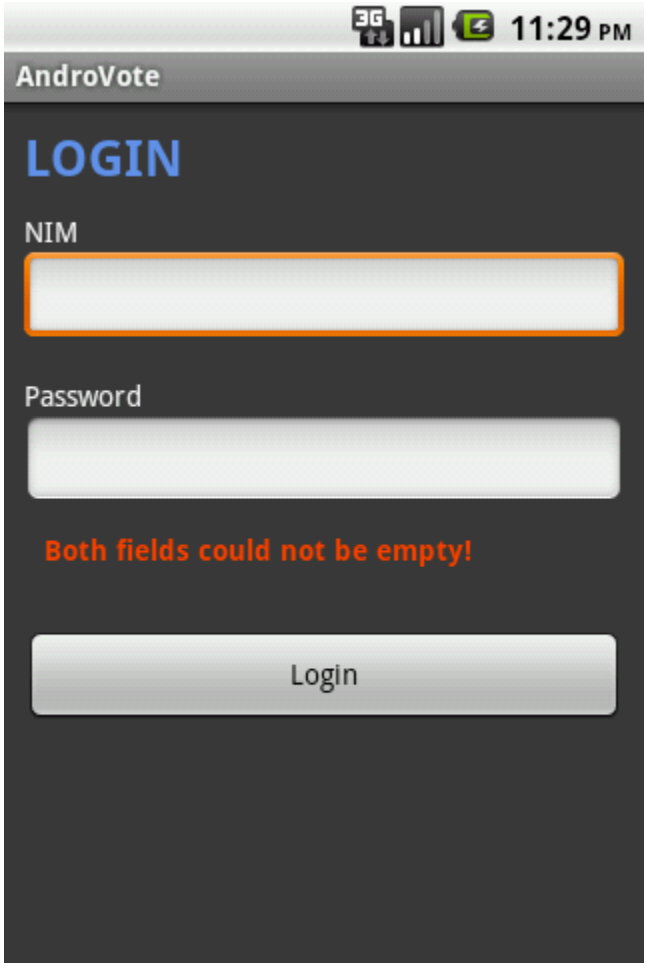
1. Tampilan Log-in

Saat pertama kali membuka aplikasi, sistem akan menampilkan halaman *login* dimana user memasukan NIM(Nomor Induk Mahasiswa) dan Password yang sesuai dengan data mahasiswa yang ada pada *database*.



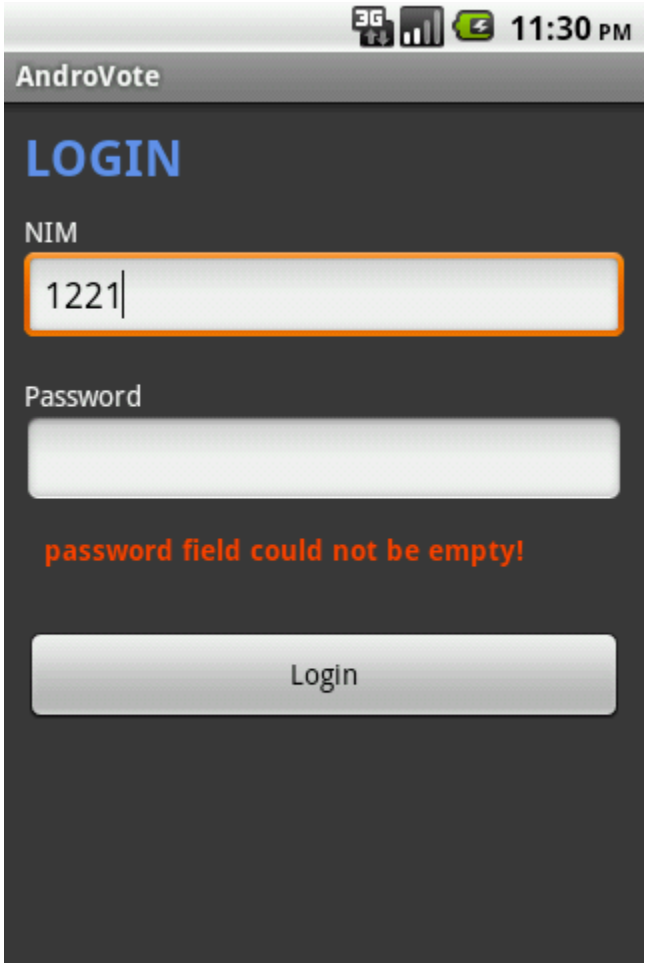
Gambar 5.1. Halaman Log-in

Jika *user*, menekan tombol *login* dengan kolom NIM dan *password* masih dalam keadaan kosong, maka sistem akan memberikan pemberitahuan kesalahan data, seperti yang digambarkan pada Gambar 5.2. di bawah ini.



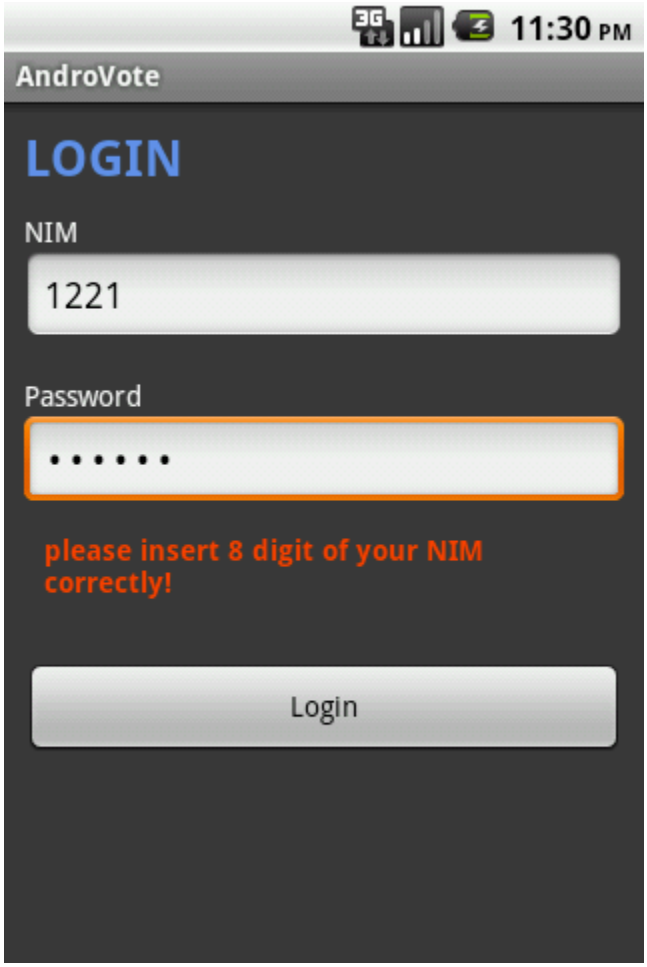
Gambar 5.2. Kesalahan *login*

Sedangkan, jika *user* hanya memasukan data pada kolom NIM lalu menekan tombol *login*, maka sistem akan memberikan reaksi seperti yang digambarkan pada Gambar 5.3. di bawah ini.



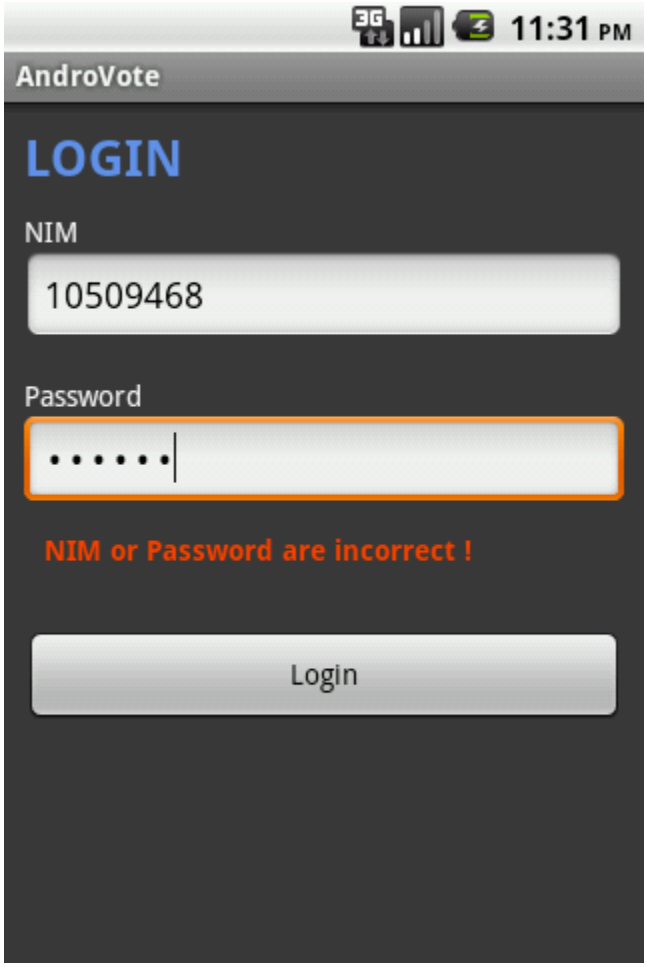
Gambar 5.3. Kesalahan *login* 2

Sedangkan, jika *user* sudah mengisi kedua kolom NIM dan *password*, tetapi jumlah NIM kurang dari 8 angka, maka sistem akan memberikan pemberitahuan seperti yang digambarkan pada Gambar 5.4.



Gambar 5.4. Kesalahan *login* 3

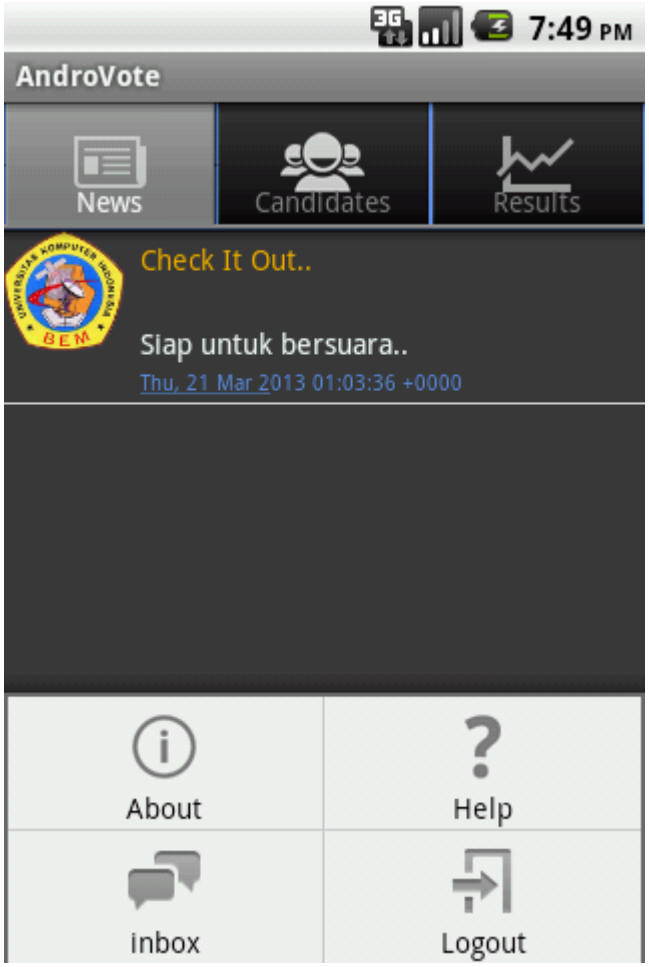
Jika ketiga kondisi yang telah disebutkan diatas telah terpenuhi, maka sistem akan mencari data masukan *login* ke dalam database, jika terdapat ketidak cocokan baik antara NIM dan *password*, maka sistem akan memberikan reaksi seperti yang digambarkan pada Gambar 5.5.



Gambar 5.5. Kesalahan *login* 4

2. Menu *Latest News Feed*

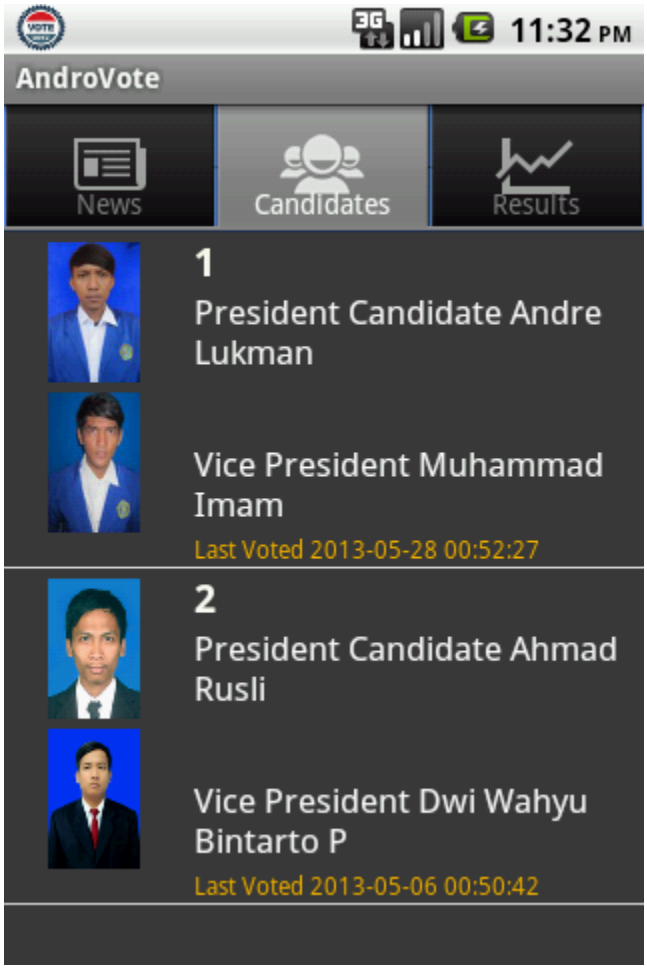
Dalam menu ini, aplikasi akan mengambil *feed* dari website Badan Eksekutif Mahasiswa (BEM) UNIKOM yaitu www.bem.unikom.ac.id .



Gambar 5.6. *Latest News Feed*

3. Menu Daftar Kandidat Presiden dan Wakil Presiden Mahasiswa

Menu ini menampilkan daftar keseluruhan daftar kandidat Presiden dan Wakil Presiden Mahasiswa.



Gambar 5.7. Daftar Kandidat Presiden dan Wakil Presiden Mahasiswa

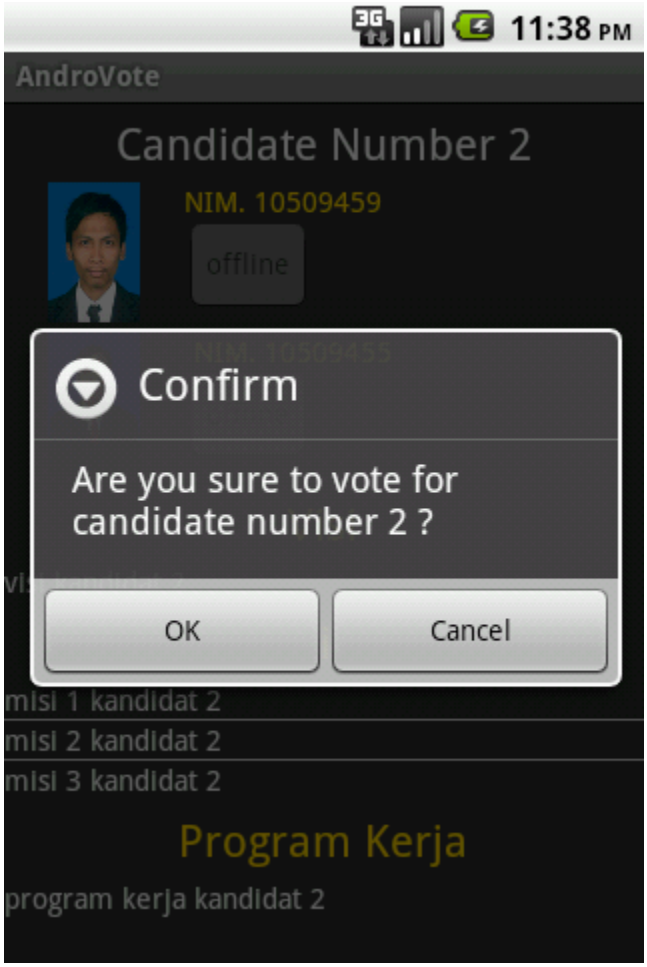
4. Halaman Detail Kandidat

Setelah user, memilih salah satu kandidat di menu kandidat, maka tampilan akan dipindahkan ke halaman datail kandidat, yang berisi foto kandidat Presiden dan Wakil Presiden Mahasiswa, NIM kandidat Presiden dan Wakil Presiden Mahasiswa, Visi, Misi, Program Kerja, Status *online* atau *offline* dari kandidat tersebut, dari halaman ini pula, mahasiswa dapat langsung melakukan pemilihan, dengan menekan tombol menu pada perangkat *android* nya, lalu kemudian menekan tombol *vote* untuk melakukan pemilihan.



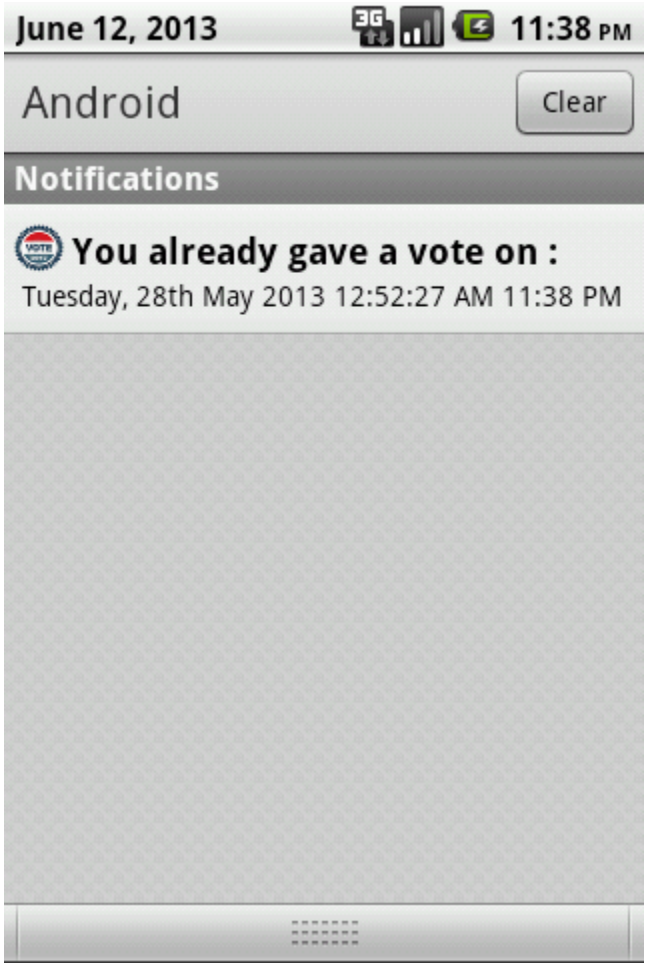
Gambar 5.8. Halaman Detail Kandidat

Setelah menekan tombol “Vote” maka sistem akan menampilkan pesan konfirmasi seperti yang digambarkan oleh Gambar 5.9. di bawah ini.



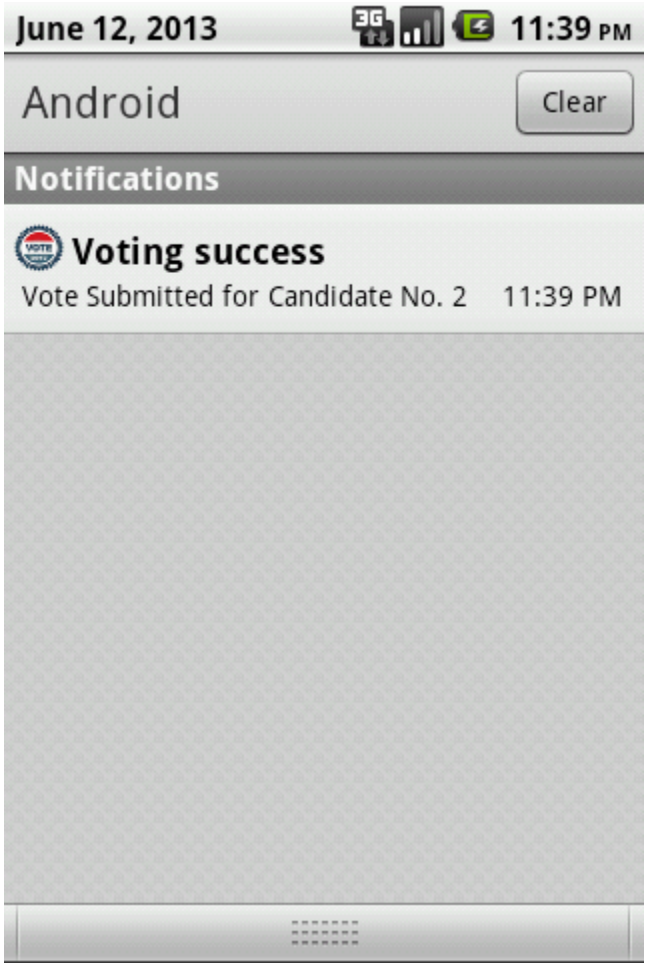
Gambar 5.9. Konfirmasi *Voting*

Jika, *user* sudah pernah memilih sebelumnya, maka sistem akan menampilkan pesan pemberitahuan seperti yang digambarkan pada Gambar 5.10 . di bawah ini.



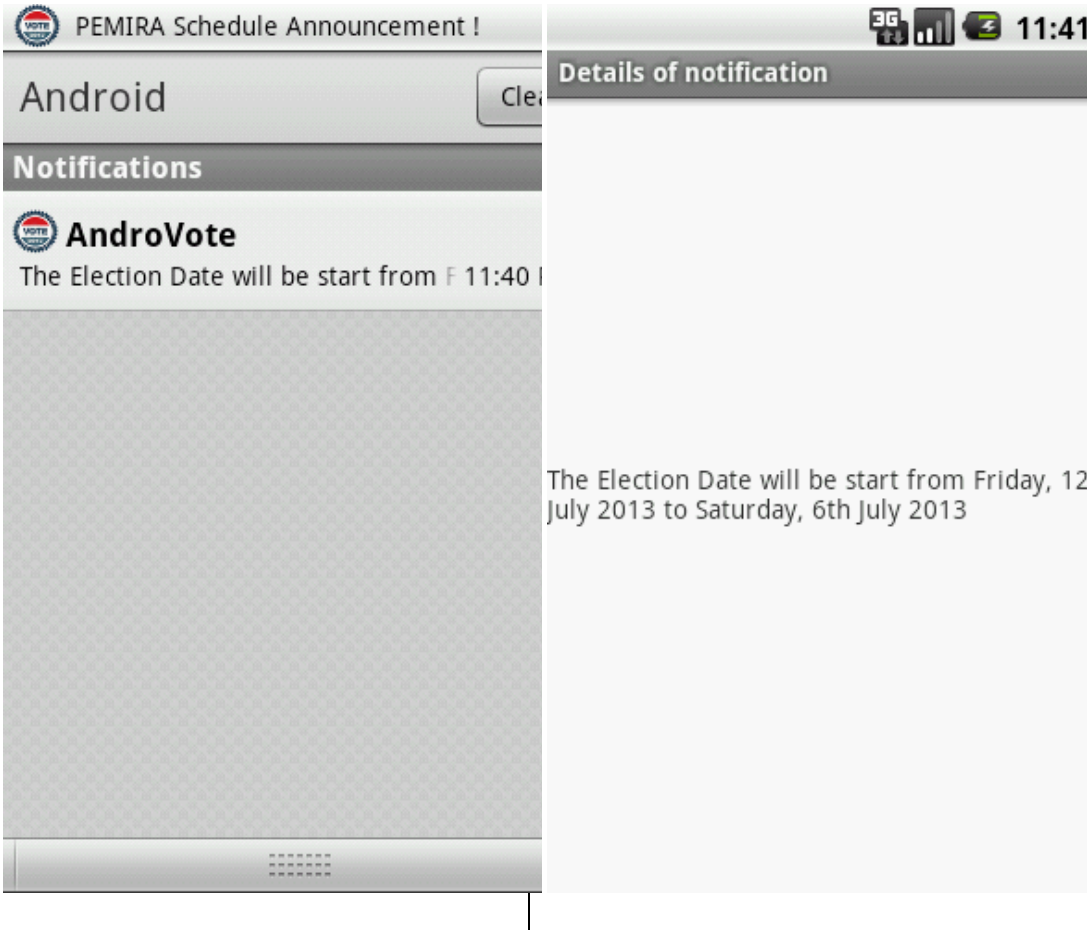
Gambar 5.10. Pesan Konfirmasi *Voting* Gagal

Akan tetapi, jika belum pernah memilih, maka sistem akan mnampilkan pesan pemberitahuan seperti yang digambarkan pada Gambar 5.11. di bawah ini.



Gambar 5.11. Pesan Konfirmasi *Voting* Sukses

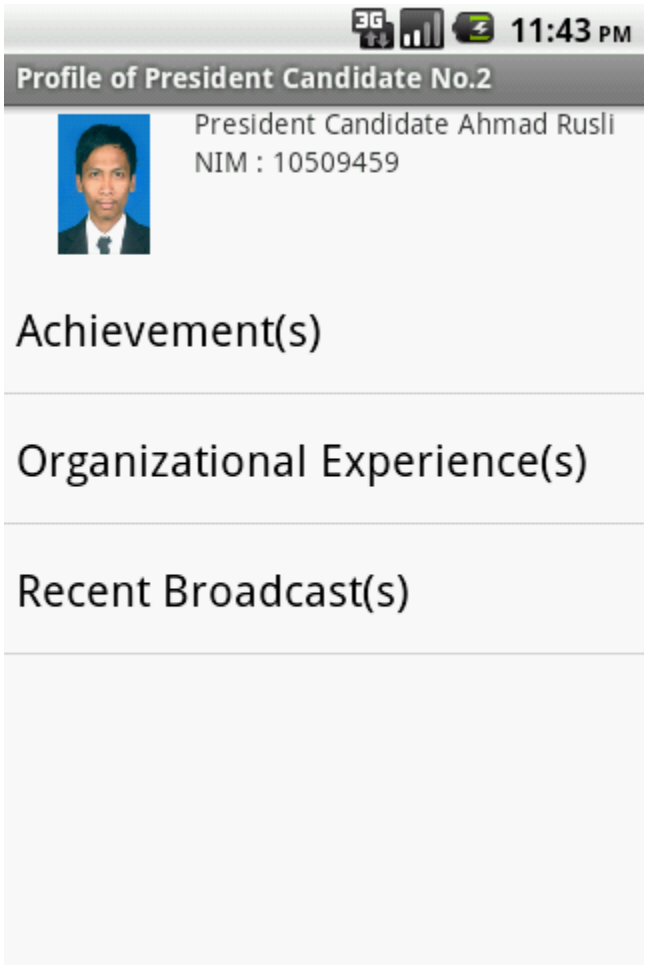
Menu *voting* akan diaktifkan dengan disesuaikan dengan jadwal pemilihan yang telah ditentukan oleh pihak panitian PEMIRA, jika user menekan tombol menu pada saat belum memasuki jadwal pemilihan, maka sistem akan memberikan pemberitahuan seperti yang digambarkan pada gambar 5.11. di bawah ini.



Gambar 5.12. Pengumuman Jadwal Pemilu

5. Halaman Profil Kandidat

User dapat mengakses halaman ini, dengan cara meng-klik salah satu foto kandidat pada halaman Detail Kandidat, halaman ini berisi prestasi, pengalaman organisasi dan pengumumuman – pengumuman yang diberikan oleh kandidat.



Gambar 5.13. Halaman Profil Kandidat

6. Halaman Prestasi Kandidat

Halaman ini dapat diakses setelah user memilih, "*Achievement(s)*" pada Profil Kandidat.



Gambar 5.14. Halaman Prestasi Kandidat

7. Halaman Pengalaman Organisasi Kandidat

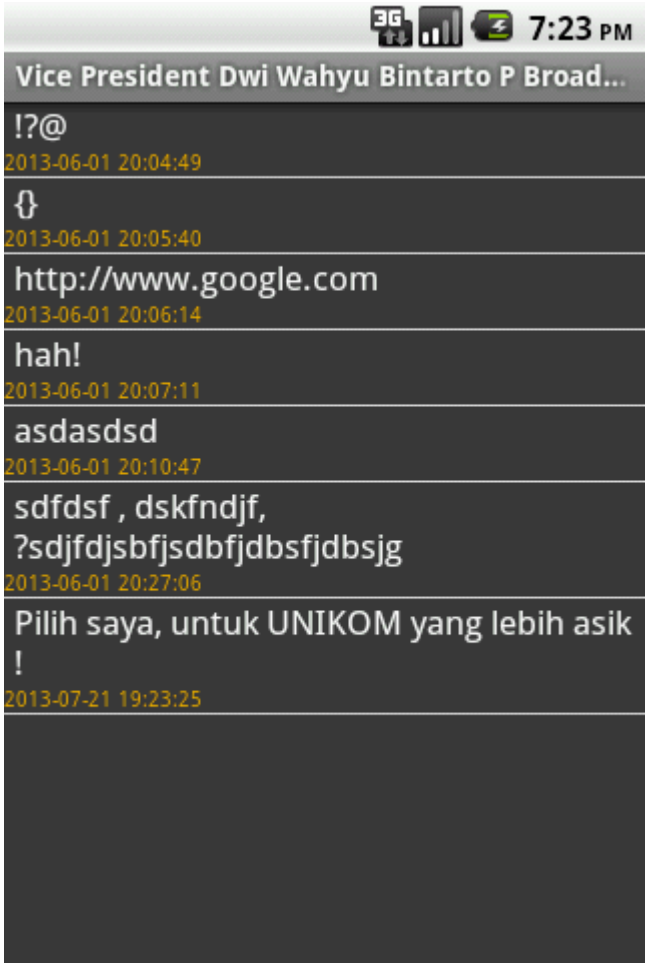
Halaman ini dapat diakses setelah user memilih, "*Organizational Experience(s)*" pada Profil Kandidat.



Gambar 5.15. Halaman Pengalaman Organisasi Kandidat

8. Halaman *Recent Broadcast(s)*

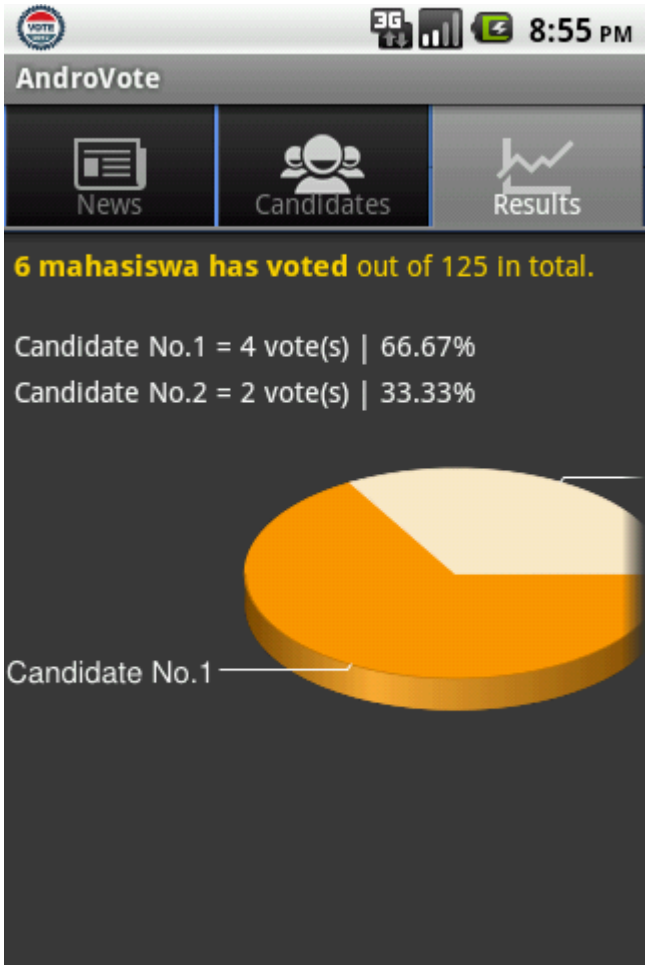
Menu ini berfungsi untuk menampilkan orasi – orasi yang disampaikan oleh masing – masing kandidat.



Gambar 5.16. Halaman *Recent Broadcast(s)* Kandidat

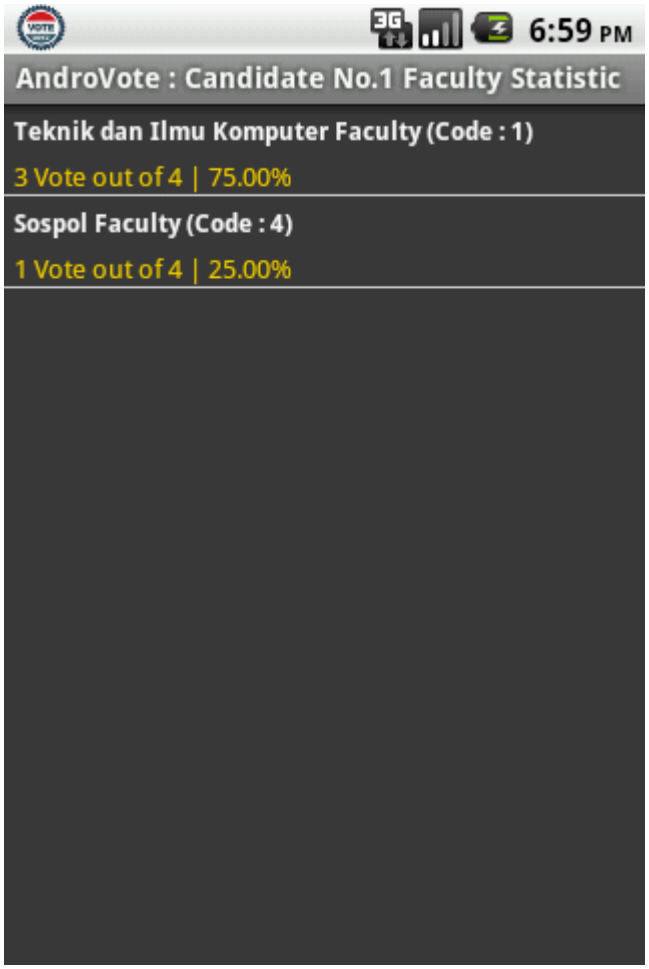
9. Menu Hasil Perolehan Suara

Menu ini menampilkan jumlah mahasiswa yang sudah melakukan pemilihan, dan juga jumlah hasil perolehan suara masing - masing kandidat.



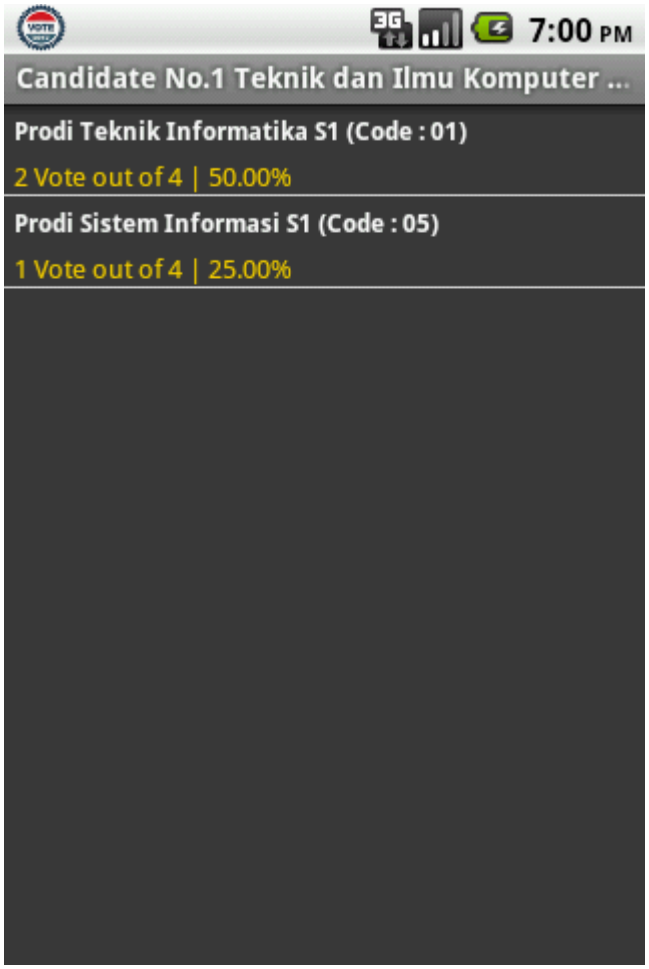
Gambar 5.17. Menu Hasil Perolehan Suara

Hasil pemilihan ini akan ditampikan sesuai dengan tanggal yang telah ditentukan oleh pihak panitia. Kemudian jika kita meng-klik salah satu kandidat maka akan ditampilkan data statistik dari fakultas dan jurusan mana saja suara yang didapatkan kandidat tersebut.



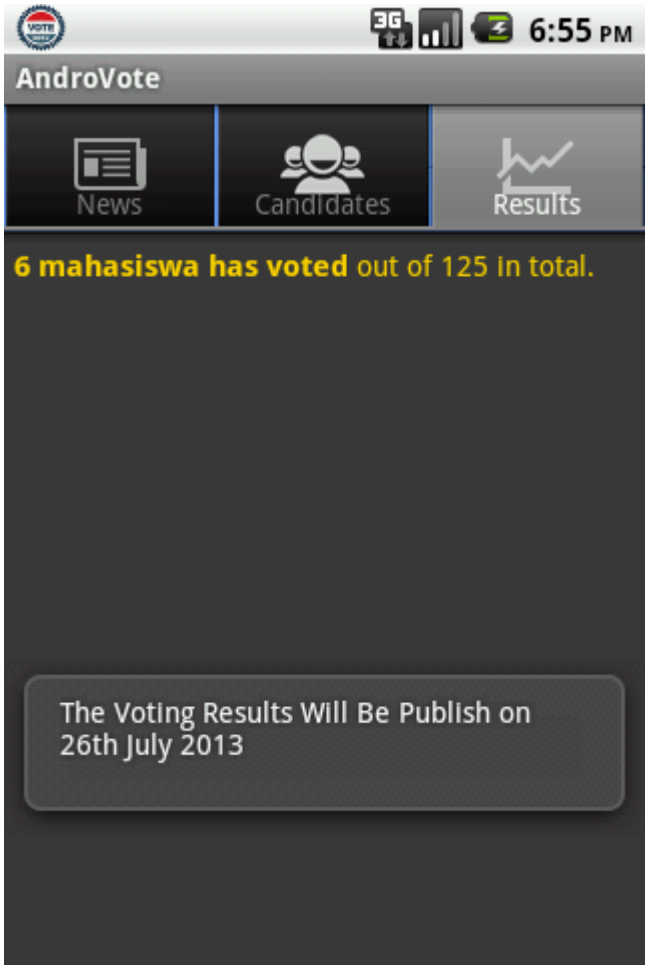
Gambar 5.18. Data Suara Kandidat Berdasarkan Fakultas

Jika kita memilih salah satu dari daftar fakultas, maka akan ditampilkan daftar jurusan dari fakultas yang dipilih yang berkontribusi pada hasil perolehan suara kandidat tersebut.



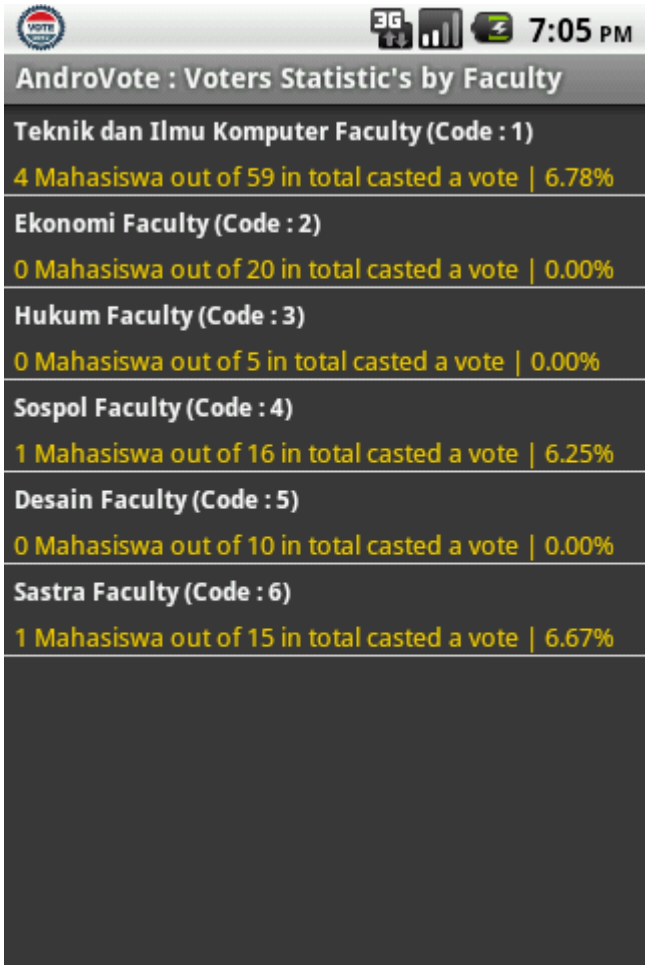
Gambar 5.19. Data Suara Kandidat Berdasarkan Jurusan

Jika *user* membuka halaman ini tidak pada saat jadwal pengumuman pemilu, maka yang akan ditampilkan hanya informasi berapa banyak mahasiswa yang sudah memilih saja.



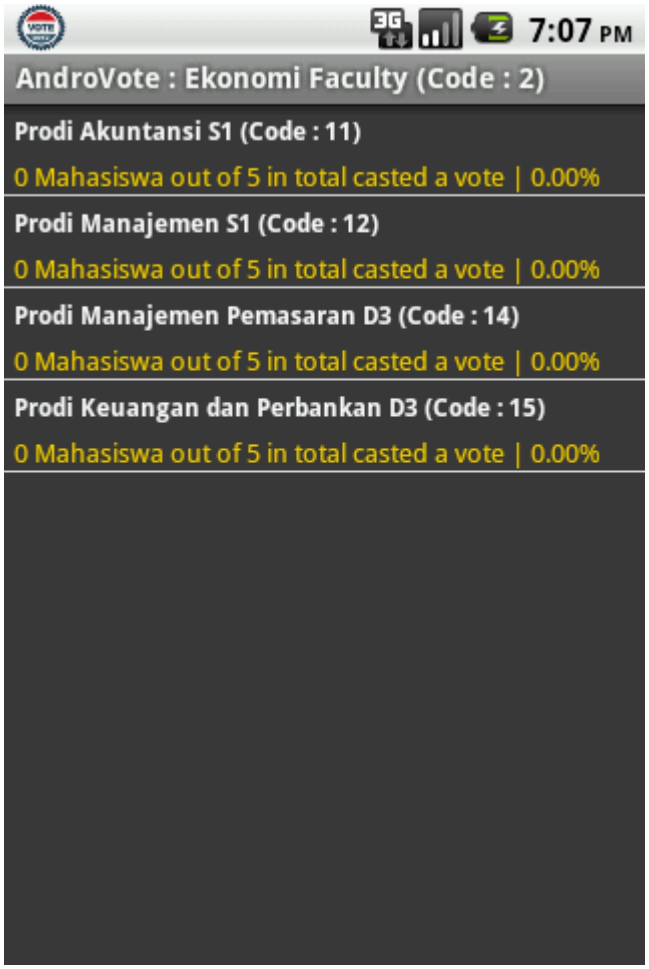
Gambar 5.20. Jumlah Mahasiswa yang sudah memilih

Kemudian, jika kita mng-klik tulisan yang bercetak tebal, maka akan menampilkan data mahasiswa yang sudah memilih berdasarkan Fakultasnya.



Gambar 5.21. Jumlah Mahasiswa yang sudah memilih berdasarkan Fakultas

Lalu, jika *user* meng-klik salah satu dari daftar fakultas tersebut, maka akan ditampilkan data mahasiswa yang sudah memilih berdasarkan jurusan dari fakultas yang dipilih oleh *user*.



Gambar 5.22. Jumlah Mahasiswa yang sudah memilih berdasarkan Jurusan

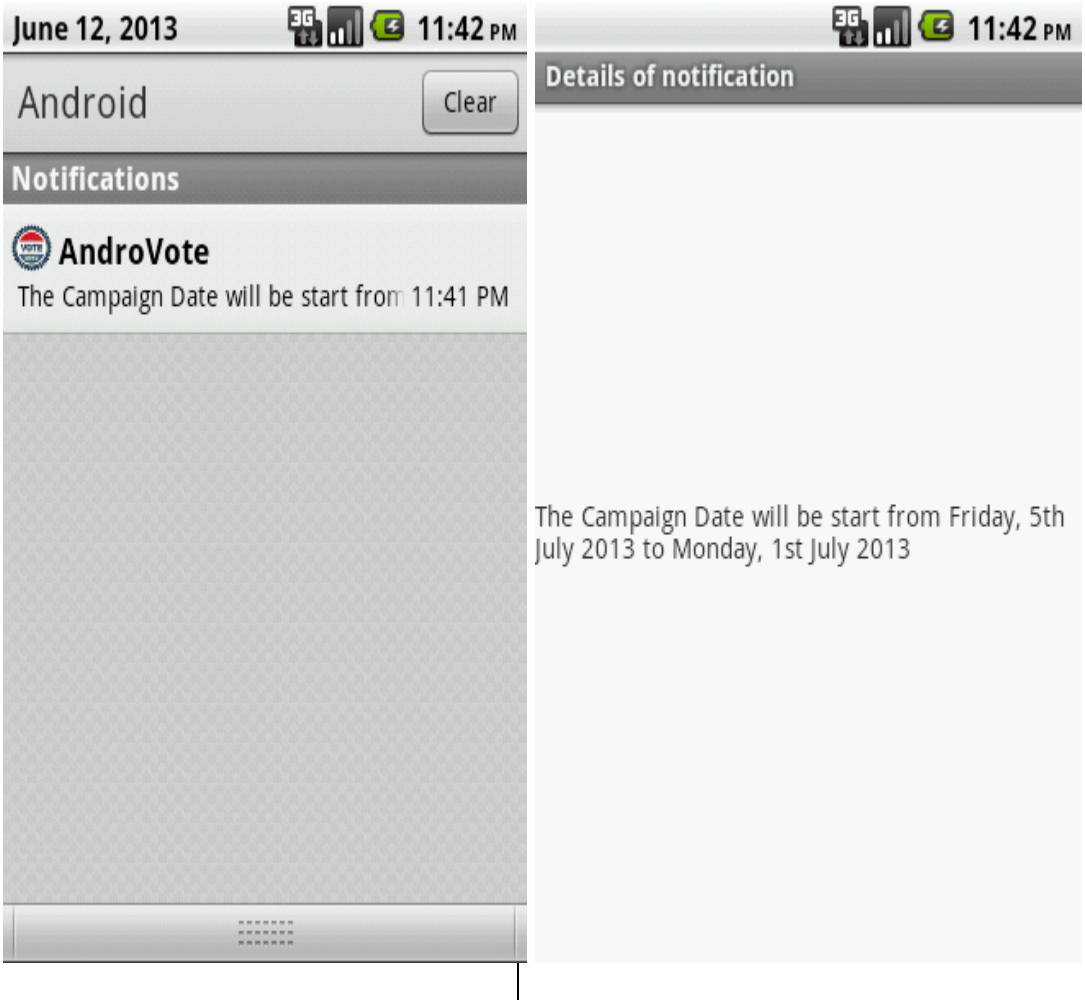
10. Menu Chatting

Menu ini, bertujuan untuk memberikan fasilitas bagi mahasiswa dan para kandidat untuk saling berinteraksi dan bertukar pendapat. Dengan menekan tombol Chat, menandakan bahawa kandidat tersebut telah *online*.



Gambar 5.23. Menu *Chatting*

Fitur *chatting* ini akan diaktifkan apabila jadwal Pemilihan Umum Raya telah memasuki masa Kampanye, jika masa Kampanye belum dimulai, maka *user* akan diberikan pemberitahuan seperti yang digambarkan Gambar 5.10. di bawah ini.



Gambar 5.24. Pemberitahuan Jadwal Kampanye

11. Halaman *Chatting*

Setelah menekan tombol *Chat*, maka tampilan halaman chatting akan ditampikan, kemudian antara kandidat dan para konstituen yang dalam hal ini adalah mahasiswa dapat mulai untuk berinteraksi.



Gambar 5.25. Halaman Chatting

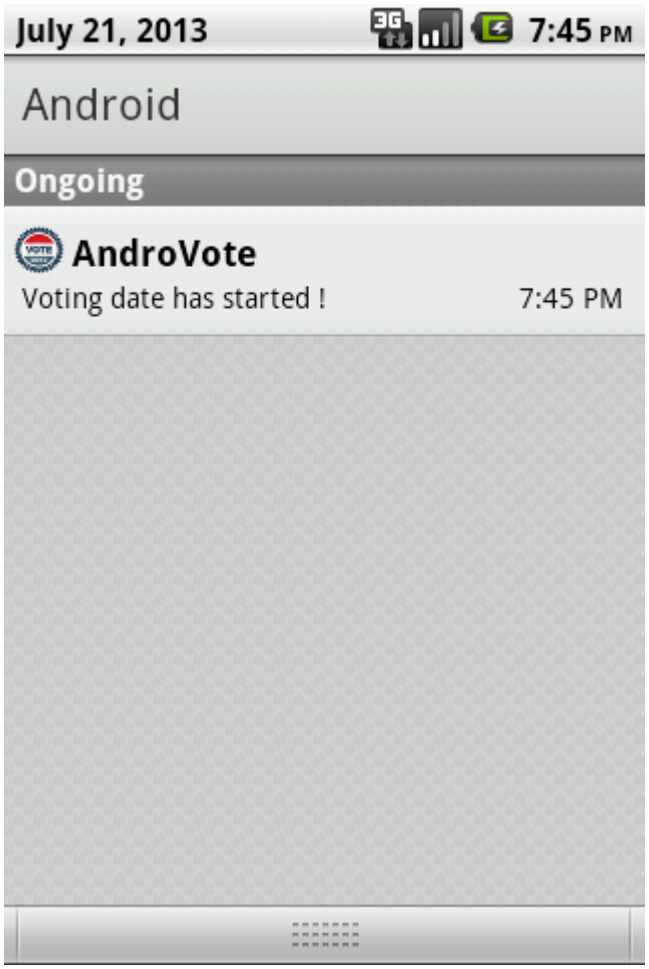
Sedangkan dari sisi kandidat, tombol *chat* akan berubah fungsi menjadi tombol *broadcast*, yang memfasilitasi kandidat untuk berorasi secara *online*, setelah meng-klik tombol *broadcast* maka akan ditampilkan pop-up menu, dengan maksimal 200 karakter yang bisa di orasikan oleh kandidat tersebut.



Gambar 5.26. Fasilitas *Brodcast*

11. Notifikasi Jadwal Pemilihan Umum Raya (PEMIRA)

Didalam aplikasi ini, jadwal yang difasilitasi di dalam aplikasi ini, hanya terbatas pada jadwal kampanye, jadwal pemilihan, dan jadwal pengumuman hasil perolehan suara. Pada teknisnya, jika tanggal awal pada tabel sosiliasasi_pemira yang terdapat pada *database* bemevoting.sql sama dengan tanggal *server*, maka notifikasi akan ditampilkan secara otomatis pada *device* user, meskipun *user* sedang tidak membuka aplikasi *AndroVote*. Gambar 5.27. menggambarkan pemberitahuan bahwa *user* bisa melakukan pemilihan suara.

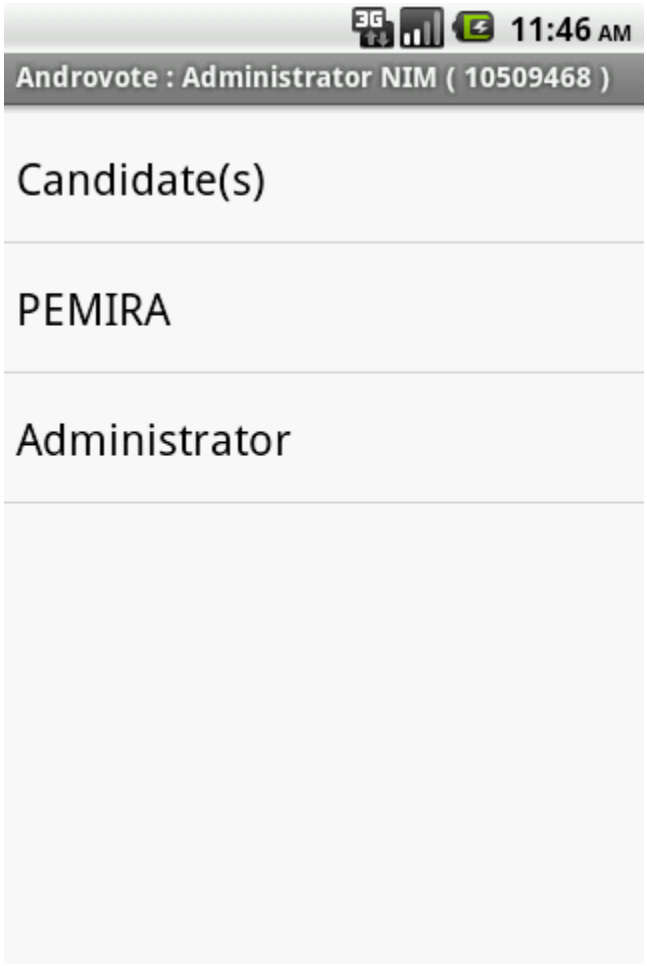


Gambar 5.27. Notifikasi Jadwal Pemilihan Umum Raya (PEMIRA)

12. Halaman *Admin*

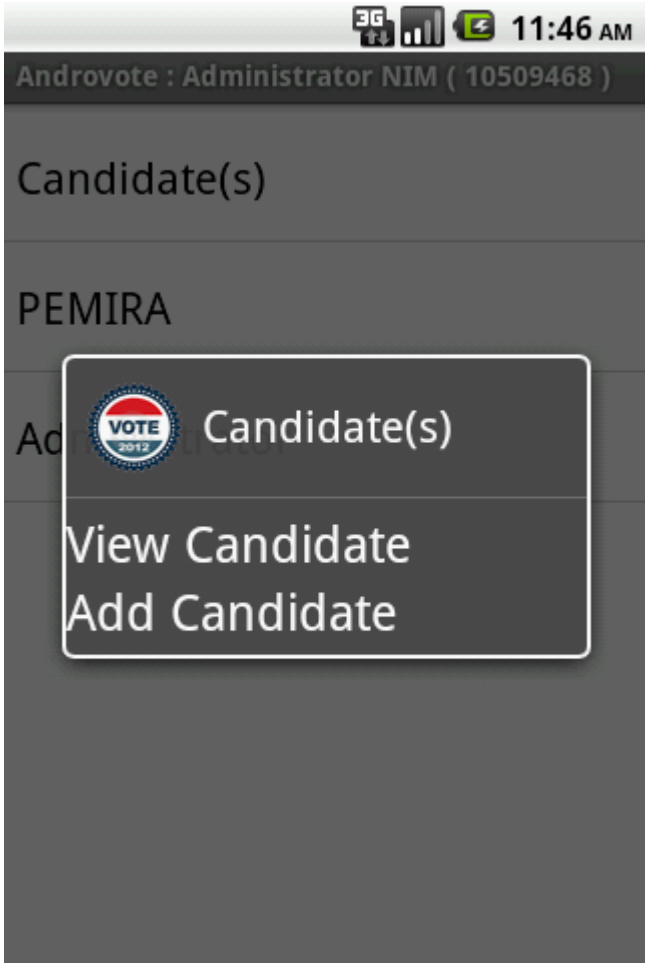
Terdapat 3 menu utama yang terdapat pada halaman admin ini, yaitu menu *Candidate(s)* yang memfasilitasi pihak kepanitiaan untuk menambahkan data kandidat baru beserta detail lainnya seperti visi, misi, program kerja, prestasi pengalaman organisasi, dan foto kandidat. Menu selanjutnya adalah menu PEMIRA yang berfungsi untuk memfasilitasi panitia untuk menentukan jadwal PEMIRA mulai dari tanggal kampanye, pemilihan dan pengumuman hasil perolehan suara. Menu yang terakhir adalah menu *Administrator* yang difungsikan

untuk membantu pihak kepanitiaan untuk menambah dan melihat panitia yang memiliki hak akses untuk mengolah data PEMIRA di aplikasi ini.



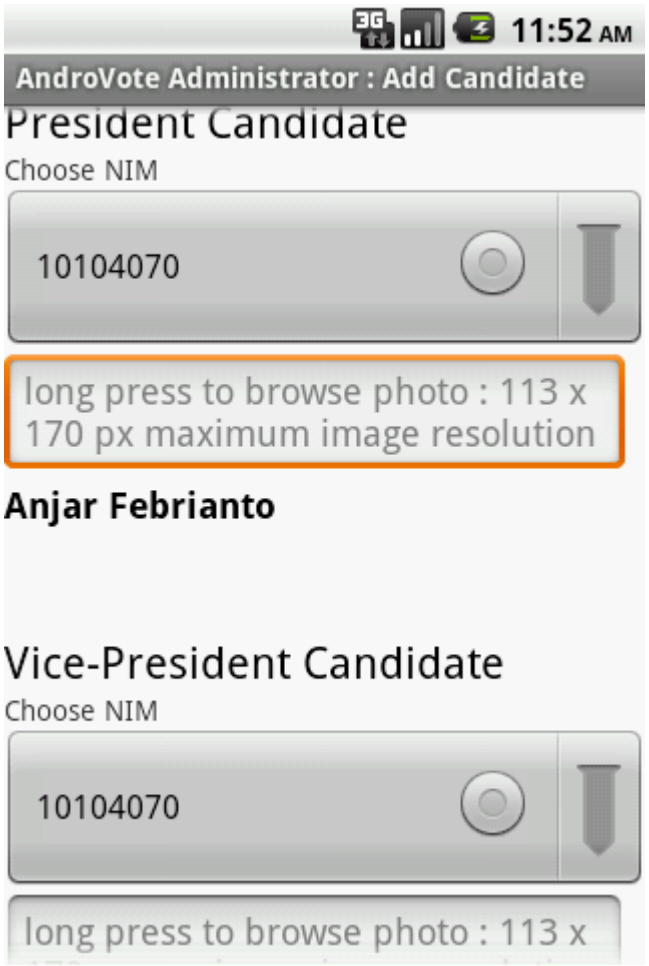
Gambar 5.28. Halaman Admin

Untuk dapat menambahkan data kandidat, panitia memilih menu *Candidate(s)*.



Gambar 5.29. Menu *Candidate(s)*

Lalu, memilih *Add Candidate*, maka kemudian sistem akan menampilkan halaman untuk menambahkan kandidat.



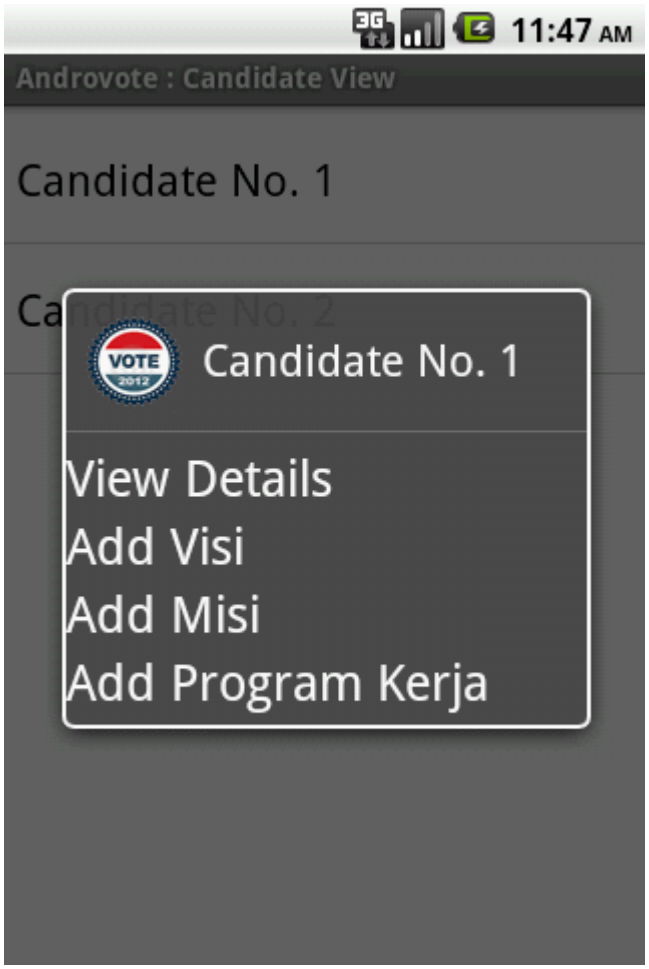
Gambar 5.30. Halaman Tambah Kandidat

Pertama – tama, admin bisa langsung memilih nim dari kandidat, setelah nim dipilih, nama dari nim tersebut akan langsung ditampilkan secara otomatis, kemudian untuk menambahkan foto admin, melakukan *long press* pada *textfield*, yang kemudian akan membaca sdcard atau memori eksternal dari *smartphone android.*



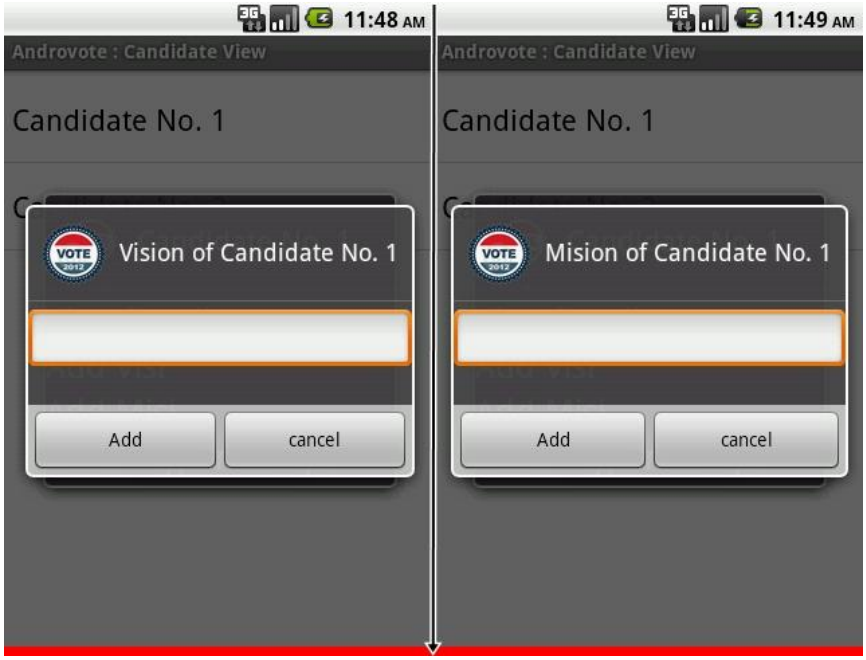
Gambar 5.31. *Browse Photo* dari SdCard

Setelah itu, admin dapat memilih foto mana yang akan dijadikan sebagai foto profil dari masing – masing kandidat. Setelah semua sudah ditentukan, admin bisa menekan tombol submit untuk menyimpan data kandidat tersebut ke *database*. Kemudian untuk menambahkan detil seperti visi, misi dan program kerja, pada menu *Candidate(s)*, admin dapat memilih *View Candidate* terlebih dahulu, lalu sistem akan menampilkan data kandidat yang ada pada *database.*

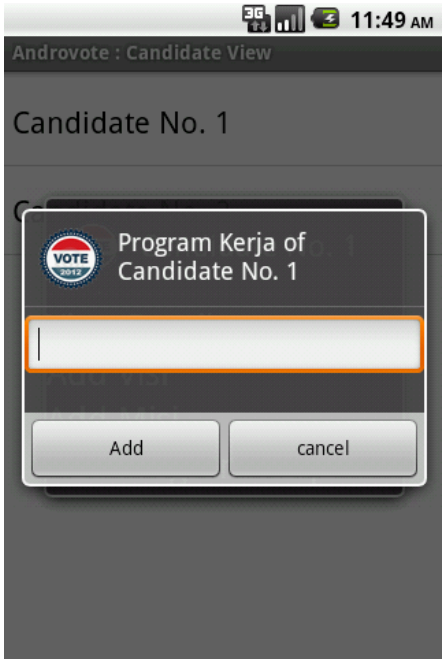


Gambar 5.32. Tambah Visi, Misi dan Program Kerja

Lalu, pilih kandidat mana yang akan ditambahkan Visi, Misi dan Program Kerjanya.



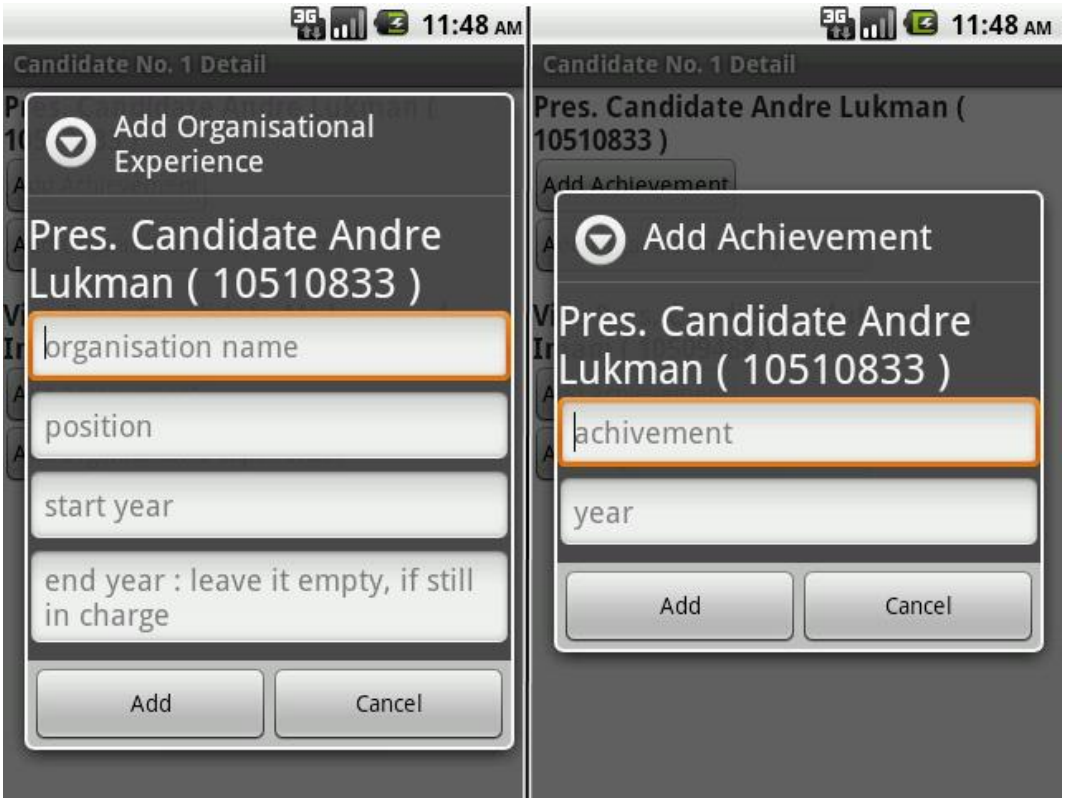
(a)



(b)

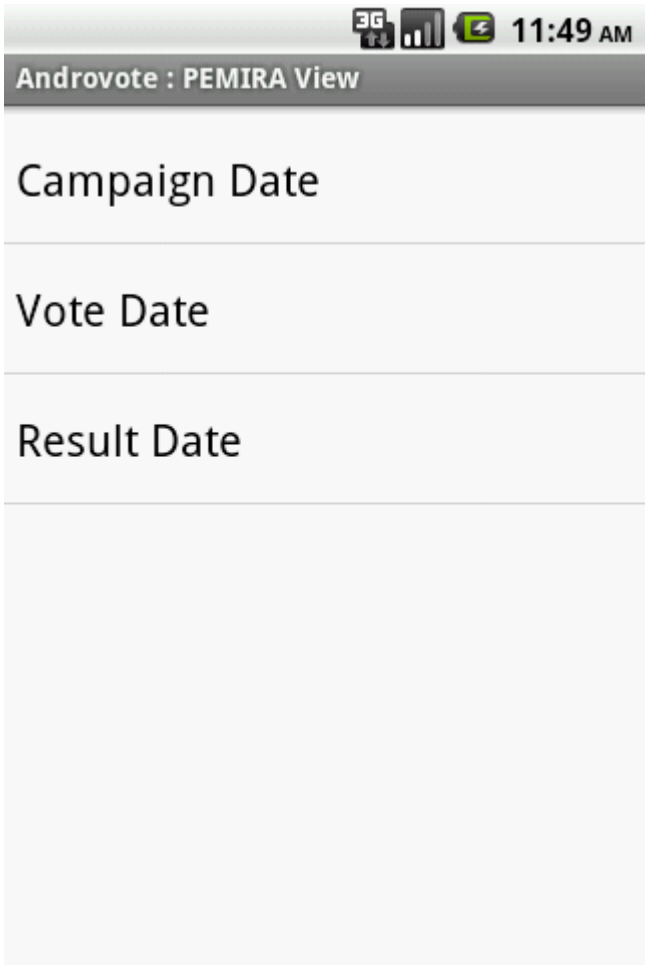
Gambar 5.33. Tampilan Tambah Visi, Misi dan Program Kerja

Sedangkan *View Details* berfungsi untuk menambahkan informasi detil dari masing – masing kandidat, seperti pengalaman organisasi dan prestasi.



Gambar 5.34. Tampilan Tambah Prestasi dan Pengalaman Organisasi

Untuk mengatur jadwal Pemilihan Umum Raya, yang dalam aplikasi ini hanya mendukung Tanggal Kampanye, Pemilihan, dan Hasil Perolehan Suara, admin memilih menu PEMIRA pada halaman utama admin.



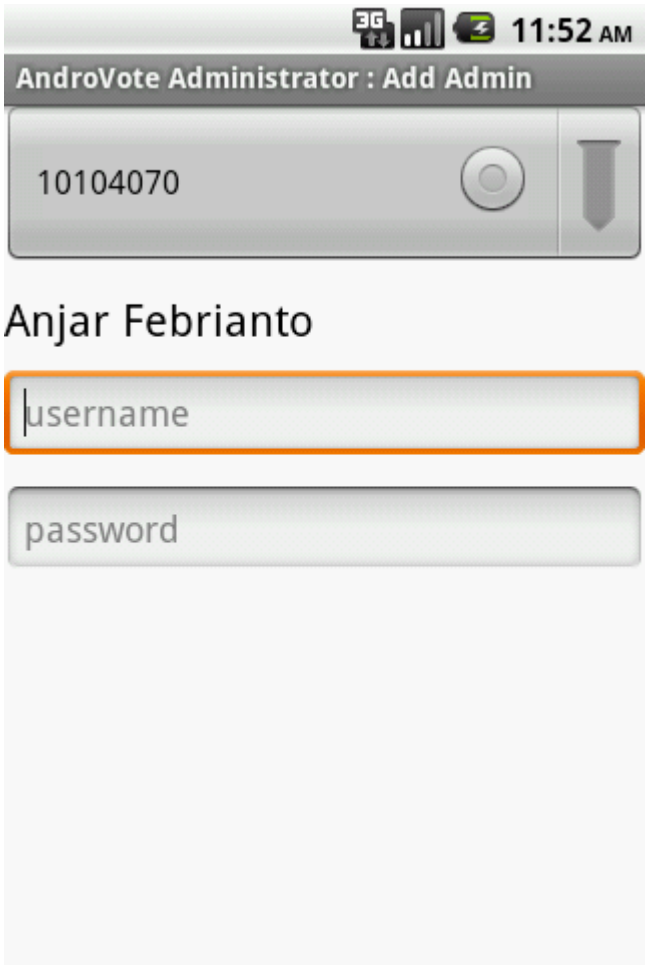
Gambar 5.35. Jadwal Pemilihan Umum Raya

Lalu, Pilih jadwal mana yang akan diatur. Gambar 5.36 menggambarkan pengaturan jadwal kampanye, klik *Set New Date* untuk mengatur jadwal kampanye yang baru.



Gambar 5.36. Atur Jadwal Pemilihan Umum Raya (Kampanye)

Kemudian tekan *Set Date* untuk menyimpan tanggal. Menu yang terakhir, adalah menu untuk menambah admin baru, pada Halaman Utama Admin pilih menu *Administrator(s),* lalu pilih *Add Admin.*



Gambar 5.37. Halaman Tambah Admin

Setelah semua data terisi, maka akan muncul tombol *Submit*, tekan tombol *Submit* untuk menyimpan data admin ke *database*.

5.2. Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Metode yang digunakan dalam pengujian ini adalah pengujian *blackbox* yang berfokus pada persyaratan fungsional dari sistem yang dibangun.

5.2.1. Rencana Pengujian

Adapun rancangan pengujian sistem yang akan diuji dengan teknik pengujian BlackBox akan penyusun kelompokan dalam tabel di bwah. Berikut ini adalah hasil dari pengujian dari Aplikasi *E-Voting* Pada Perangkat *Mobile* berbasis *Android.*

Tabel 5.1. Rencana Pengujian

No.	Komponen yang diuji	Skenario Butir Uji	Jenis Pengujian
1	Halaman Login	Velidasi Kolom Nim	*Black-Box*
		Validasi Kolom Password	*Black-Box*
		Verifikasi Nim dan Password dengan Database.	*Black-Box*
2	Halaman Menu Utama	Menampilkan Rss Feed dari www.bem.unikom.ac.id	*Black-Box*
		Menampilkan Halaman Daftar Kandidat	*Black-Box*
		Menampilkan Halaman Hasil Perolehan Suara	*Black-Box*
3	Halaman Detail Kandidat	Menampilkan Visi, Misi dan Program Kerja Kandidat	*Black-Box*
		Menampilkan prestasi, pengalaman organisasi, dan orasi kandidat	*Black Box*
4	Menu "*Vote*"	Mengaktifkan menu, sesuai dengan jadwal yang telah ditentukan panitia.	*Black-Box*

		Melakukan pemilihan	*Black-Box*
		Pemberitahuan voting berhasil	*Black-Box*
		Pemberitahuan gagal melakukan voting jika sudah memilih sebelumnya.	*Black-Box*
5	Halaman Perolehan Hasil Suara	Menampilkan Jumlah Mahasiswa yang sudah memilih	*Black-Box*
		Menampilkan hasil perolehan suara, sesuai dengan jadwal yang telah ditentukan oleh panitia	*Black-Box*
		Menampilkan data mahasiswa yang sudah memilih berdasarkan fakultas dan jurusannya	*Black Box*
		Menampilkan data suara perolahan kandidat berdasarkan fakultas dan jurusannya	*Black Box*
6	Tombol "*Chat*"	Mengaktifkan tombol, sesuai dengan status online / tidaknya kandidat	*Black-Box*
		Chatting antara kandidat dengan mahasiswa	*Black-Box*
		Memberikan notifikasi jika ada pesan masuk	*Black-Box*
		Mengganti tombol "*Chat*" menjadi tombol "*Broadcast*" jika kandidat	*Black-Box*

		mengakses profilnya masing masing.	
7	Menu Admin	Menambah data kandidat baru	*Black-Box*
		Mengatur jadwal kegiatan PEMIRA	*Black-Box*
		Menambah dan melihat data admin	*Black Box*
8	Notifikasi Jadwal PEMIRA (Kampanye, Voting dan Hasil Perolehan Suara)	Menampilkan pemberitahuan secara otomatis pada *device user* jika tanggal kampanye / voting / pengumuman hasil perolehan suara sudah dimulai.	*Black Box*

5.2.2. Kasus dan Hasil Pengujian

Berikut ini adalah kasus untuk menguji perangkat lunak yang sudah dibangun menggunakan metode *BlackBox* berdasarkan Tabel 5.2 Rencana pengujian yang telah dibuat sebelumnya, pengujian ini di implementasikan telepon seluler seperti *smartphone* yang memiliki sistem operasi android.

Tabel 5.2. Kasus dan Hasil Uji

No.	Komponen yang diuji	Skenario Butir Uji	Jenis Pengujian
1	Halaman Login	Velidasi Kolom Nim	[√] Berhasil [ ] Gagal
		Validasi Kolom Password	[√] Berhasil [ ] Gagal

		Verifikasi Nim dan Password dengan Database.	[√] Berhasil [ ] Gagal
2	Halaman Menu Utama	Menampilkan Rss Feed dari www.bem.unikom.ac.id	[√] Berhasil [ ] Gagal
		Menampilkan Halaman Daftar Kandidat	[√] Berhasil [ ] Gagal
		Menampilkan Halaman Hasil Perolehan Suara	[√] Berhasil [ ] Gagal
3	Halaman Detail Kandidat	Menampilkan Visi, Misi dan Program Kerja Kandidat	[√] Berhasil [ ] Gagal
		Menampilkan prestasi, pengalaman organisasi, dan orasi kandidat	[√] Berhasil [ ] Gagal
4	Menu "*Vote*"	Mengaktifkan menu, sesuai dengan jadwal yang telah ditentukan panitia.	[√] Berhasil [ ] Gagal
		Melakukan pemilihan	[√] Berhasil [ ] Gagal
		Pemberitahuan voting berhasil	[√] Berhasil [ ] Gagal
		Pemberitahuan gagal melakukan voting jika sudah memilih sebelumnya.	[√] Berhasil [ ] Gagal
5	Halaman Perolehan Hasil Suara	Menampilkan Jumlah Mahasiswa yang sudah memilih	[√] Berhasil [ ] Gagal

		Menampilkan hasil perolehan suara, sesuai dengan jadwal yang telah ditentukan oleh panitia	[√] Berhasil [ ] Gagal
		Menampilkan data mahasiswa yang sudah memilih berdasarkan fakultas dan jurusannya	[√] Berhasil [ ] Gagal
		Menampilkan data suara perolahan kandidat berdasarkan fakultas dan jurusannya	[√] Berhasil [ ] Gagal
6	Tombol "*Chat*"	Mengaktifkan tombol, sesuai dengan status online / tidaknya kandidat	[√] Berhasil [ ] Gagal
		Chatting antara kandidat dengan mahasiswa	[√] Berhasil [ ] Gagal
		Memberikan notifikasi jika ada pesan masuk	[√] Berhasil [ ] Gagal
		Mengganti tombol "*Chat*" menjadi tombol "*Broadcast*" jika kandidat mengakses profilnya masing masing.	[√] Berhasil [ ] Gagal
7	Menu Admin	Menambah data kandidat baru	[√] Berhasil [ ] Gagal
		Mengatur jadwal kegiatan PEMIRA	[√] Berhasil [ ] Gagal
		Menambah dan melihat data admin	[√] Berhasil

			[ ] Gagal
8	Notifikasi Jadwal PEMIRA (Kampanye, Voting dan Hasil Perolehan Suara)	Menampilkan pemberitahuan secara otomatis pada *device user* jika tanggal kampanye / voting / pengumuman hasil perolehan suara sudah dimulai.	[√] Berhasil [ ] Gagal

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus uji *sample* di atas, dapat ditarik kesimpulan bahwa secara fungsional, perangkat lunak mengeluarkan hasil yang sesuai dengan yang diharapkan.