

**SISTEM INFORMASI E-VOTING KETUA OSIS  
SMAN 1 PAKUSARI JEMBER**

**LAPORAN AKHIR**



oleh

**Aditya Dwi Kurniawan  
NIM E31151772**

**PROGRAM STUDI MANAJEMEN INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI JEMBER  
2018**

**SISTEM INFORMASI E-VOTING KETUA OSIS  
SMAN 1 PAKUSARI JEMBER**

**LAPORAN AKHIR**



Sebagai salah satu syarat untuk memperoleh gelar Ahli Madya(A.Md.)  
Di Program Studi Manajemen Informatika  
Jurusan Teknologi Informasi

oleh

**Aditya Dwi Kurniawan  
NIM E31151772**

**PROGRAM STUDI MANAJEMEN INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI JEMBER  
2018**

KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI  
POLITEKNIK NEGERI JEMBER

SISTEM INFORMASI E-VOTING KETUA OSIS  
SMAN 1 PAKUSARI JEMBER

Aditya Dwi Kurniawan (E31151772)  
Telah Diuji pada Tanggal 10 Januari 2018  
Telah Dinyatakan Memenuhi Syarat

HALAMAN PENGESAHAN

Ketua Penguji,

I Putu Dody Lesmana, ST, MT  
NIP. 19790921200501 1 001

Sekretaris Penguji,

Nugroho Setyo Wibowo, ST, MT  
NIP. 19740519 200312 1 002

Anggota Penguji,

Wahyu Kurnia Dewanto, S.Kom, MT  
NIP. 19710408 2002112 1 003

Dosen Pembimbing I

I Putu Dody Lesmana, ST, MT  
NIP. 19790921200501 1 001

Dosen Pembimbing II

Nugroho Setyo Wibowo, ST, MT  
NIP. 19740519 200312 1 002



Menyetujui  
Ketua Jurusan Teknologi Informasi

Wahyu Kurnia Dewanto, S.Kom, MT  
NIP. 19710408 2002112 1 003

## **SURAT PERNYATAAN**

Saya yang bertanda tangan di bawah ini:

Nama : Aditya Dwi Kurniawan

NIM : E31151772

Menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam laporan akhir saya yang berjudul “SISTEM INFORMASI E-VOTING KETUA OSIS SMAN 1 PAKUSARI JEMBER” merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing, dan belum pernah diajukan dalam bentuk apa pun pada perguruan tinggi mana pun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan dari penulis lain telah disebutkan dalam naskah dan dicantumkan dalam daftar pustaka di bagian akhir Laporan Akhir ini.

Jember, 10 January 2018

Aditya Dwi Kurniawan  
NIM E31151772

## **HALAMAN PERSEMBAHAN**

Dengan mengucap syukur Alhamdulillah, karya sederhana ini teruntuk orang-orang terkasih :

- Orang tua tersayang, Bapak Bambang Yasminto R dan Ibu Yayuk Mariaty orang yang telah berjuang demi keluarga. Terima kasih selalu mendoakan yang terbaik untuk anak – anaknya, sehingga saya dapat menyelesaikan tugas akhir ini.
- Bapak I Putu Dody Lesmana,S.T, M.T dan Bapak Nugroho Setyo Wibowo,S.T, M.T terima kasih untuk bimbingan dan semangatnya selama penyelesaian tugas akhir ini.
- Aloysius Yosi PEMPE, Ayuni Raudlatul Jannah, Saskia Ramadhani, terima kasih telah memberi pinjaman laptop selama menyelesaikan tugas akhir saya.
- Teman - teman seperjuangan MIF 2015 Golongan D, terima kasih untuk kekompakan, bantuan, dan kerja sama kalian. Terima kasih untuk waktunya selama menempuh studi di Politeknik Negeri Jember. Kalian luar biasa.

----- TerimaKasih -----  
Almamaterku Tercinta

## **HALAMAN MOTTO**

“DON’T BE AFRAID TO FAIL, BE AFRAID NOT TO TRY”

“SUCCESS IS NO ACCIDENT. IT IS HARD WORK, PERSEVERANCE, LEARNING,  
STUDYING, SACRIFICE AND MOST OF ALL, LOVE OF WHAT YOU ARE  
DOING”  
(PELE)

“ROBBIHFIRLI, WARHAMNI, WAJBURNI, WARFA’NI, WARZUQNI,  
WAHDINI, WA”AFINII, WA’FU’ANNI “

“MAN JADDA WAJADA”

SISTEM INFORMASI E-VOTING KETUA OSIS SMAN 1 PAKUSARI JEMBER.

Pembimbing (2 orang).

**Aditya Dwi Kurniawan**

**PROGRAM STUDI MANAJEMEN INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

## **ABSTRAK**

OSIS yang sangat berperan penting bagi sekolah karena OSIS sebagai satuan atau kelompok kerjasama para siswa yang dibentuk dalam usaha mencapai tujuan bersama, oleh karena itu pemilihan ketua osis harus dilakukan secara benar, Pemilihan Ketua OSIS di SMAN 1 PAKUSARI Jember masih dilakukan secara konvesional selain itu dalam pemilihan ketua osis mengeluarkan tenaga dan memerlukan waktu yang banyak dalam menjelaskan visi misi calon serta pengumpulan dan perhitungan suara, karena panitia dan calon ketua OSIS harus keliling ke tiap-tiap kelas mulai dari kelas 10 hingga kelas 12 dan ruang guru serta perhitungan suara sehingga dapat mengganggu kegiatan belajar mengajar, Dari permasalahan diatas maka penulis mengembangkan suatu sistem untuk melaksanakan pemilihan ketua osis yang memanfaatkan teknologi informasi yang disebut Elektronic Voting (e-Voting). Hasil dari penelitian ini adalah sebuah sistem e-voting berbasis desktop menggunakan Bahasa pemerograman java, Metode yang digunakan dalam pengembangan system ini menggunakan prototype yang prosesnya meliputi communication, quick plan, modelling quick design, construction of prototype, deployment delivery and feedback, Dengan system tersebut tentunya akan sangat membantu sekolah dalam memilih ketua OSIS seperti kemudahan, efisiensi, peningkatan partisipasi pemilih, dan keakuratan penghitungan suara secara bijaksana dan menghindari kecurangan,sistem dari penulis sangat mudah,cepat,efisien dan akurat serta membuat pemilih percaya atas hasil pemilihan

**Kata Kunci** — Pemilihan Ketua OSIS, E-Voting ,Prototype

SISTEM INFORMASI E-VOTING KETUA OSIS SMAN 1 PAKUSARI JEMBER.

Pembimbing (2 orang).

**Aditya Dwi Kurniawan**

**PROGRAM STUDI MANAJEMEN INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

## ***ABSTRACT***

*OSIS which plays an important role for the school because OSIS as a unit or group of students collaboration formed in the effort of achieving the common goal, therefore the election osis chairman must be done correctly, the choice of chairman of OSIS in SMAN 1 PAKUSARI Jember is still done conventionally the election of the osis leader takes power and takes a lot of time in explaining the vision of the candidate's mission and the sound settlement, as the OSIS committee and candidate chairman must travel to each class from grade 10 to grade 12 and the teacher's room and voice counting can interfere with learning activities , From the question above, the authors build a system to carry out the election chairman of the osis utilizing information technology called Elektronic Voting (e-Voting). The result of this research is a desktop-based e-voting system using java probe, the method used in the development of this system using prototype which process communication range, quick plan, fast design modeling, prototype construction, dispatching and feedback, with the system of course will be very fostering schools in choosing the OSIS chairman such as ease, efficiency, increase voter participation, and accuracy of vote counting wisely and avoid cheating, the system from the author is very easy, fast, efficient and accurate*

***Keywords — Selection of Chairman of OSIS, E-Voting, Prototype Pemilihan Ketua OSIS, E-Voting ,Prototype***

## **RINGKASAN**

**SISTEM INFORMASI E-VOTING KETUA OSIS SMAN 1 PAKUSARI JEMBER,**  
Aditya Dwi Kurniawan, NIM E31151772, Tahun 2018, Teknologi Informasi,  
Politeknik Negeri Jember, I Putu Dody Lesmana,S.T, M.T, (Pembimbing I), dan  
Nugroho Setyo Wibowo,S.T, M.T (Pembimbing II).

Setiap sekolah memiliki ketentuan dan kebijakan masing-masing dalam menyejahterakan sekolahnya. Salah satunya OSIS yang sangat berperan penting bagi sekolah karena OSIS sebagai satuan atau kelompok kerjasama para siswa yang dibentuk dalam usaha mencapai tujuan bersama, oleh karena itu pemilihan ketua osis harus dilakukan secara benar , ketua osis yang diharapkan mampu membawa sekolah kearah yang lebih baik lagi, Pelaksanaan pemilihan harus berlangsung LUBER (Langsung Umum Bebas Rahasia). Hal ini dilakukan untuk melatih para siswa dalam menjalankan salah satu asas dalam demokrasi mulai dari pemberian kartu suara, pencoblosan, memasukan kartu suara, pencelupan atau penandaan jari serta perhitungan suara

Kemajuan teknologi yang ada saat ini terus berkembang, laju perkembangan teknologi yang sangat cepat memang tak bisa kita hindari dan hentikan,begitu juga dengan perangkat-perangkat elektronik, Hal ini semakin mempermudah manusia dalam melakukan suatu kegiatan, termasuk kegiatan pemilihan ketua OSIS

## **PRAKATA**

Puji syukur penulis panjatkan ke hadirat Allah SWT atas berkat rahmat dan karunia-Nya, maka penulisan karya tulis ilmiah berjudul SISTEM INFORMASI E-VOTING KETUA OSIS SMAN 1 PAKUSARI JEMBER dapat diselesaikan dengan baik.

Tulisan ini adalah laporan hasil penelitian yang dilaksanakan mulai bulan Juli 2017 sampai dengan Januari 2018 bertempat di Politeknik Negeri Jember, yang dilakukan sebagai salah satu syarat untuk memperoleh gelar Ahli Madya (A.Md) di Program Studi Manajemen Informatika Jurusan Teknologi Informasi Politeknik Negeri Jember.

Penulis menyampaikan penghargaan dan ucapan terima kasih yang sebesar-besarnya kepada:

1. Direktur Politeknik Negeri Jember,
2. Ketua Jurusan Teknologi Informasi,
3. Ketua Program Studi Manajemen Informatika,
4. Bapak I Putu Dody Lesmana,S.T, M.T, selaku Pembimbing I,
5. Bapak Nugroho Setyo Wibowo,S.T, M.T, selaku Pembimbing II,
6. Rekan-rekanku dan semua pihak yang telah ikut membantu dalam pelaksanaan penelitian dan penulisan laporan ini

Laporan Karya Tulis Ilmiah ini masih kurang sempurna, mengharapkan kritik dan saran yang sifatnya membangun guna perbaikan di masa mendatang. Semoga tulisan ini bermanfaat.

Jember, 10 January 2018

Aditya Dwi Kurniawan



**PERNYATAAN PERSETUJUAN PUBLIKASI  
KARYA ILMIAH UNTUK KEPENTINGAN  
AKADEMIS**

Yang bertanda tangan dibawah ini, saya:

**Nama : Aditya Dwi Kurniawan**  
**NIM : E31151772**  
**Program Studi : Manajemen Informatika**  
**Jurusan : Teknologi Informasi**

Demi pengembangan ilmu pengetahuan, saya menyetujui untuk memberikan kepada UPT. Perpustakaan Politeknik Negeri Jember, Hak Bebas Royalti NonEksklusif (Non-Exclusive Royalty Free Right) atas Karya Ilmiah berupa Laporan Tugas Akhir yang berjudul:

**SISTEM INFORMASI E-VOTING KETUA OSIS SMAN 1 PAKUSARI  
JEMBER**

Dengan Hak Bebas Royalti Non-Eksklusif ini UPT. Perpustakaan Politeknik Negeri Jember berhak menyimpan, mengalihkan media atau format, megelola dalam bentuk Pangkalan Data (DataBase), mendistribusikan karya dan menampilkan atau mempublikasikannya di internet atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis atau pencipta.

Saya bersedia untuk menanggung secara pribadi tanpa melibatkan pihak Politeknik Negeri Jember, segala bentuk tuntutan hukum yang timbul atas Pelanggaran Hak Cipta dalam Karya Ilmiah ini.

Demikian pernyataan ini saya buat dengan sebenarnya.

**Dibuat di : Jember  
Pada Tanggal: 10 Januari 2018  
Yang Menyatakan,**

**Nama : Aditya Dwi Kurniawan  
NIM : E31151772**

## DAFTAR ISI

	Halaman
<b>JUDUL .....</b>	<b>ii</b>
<b>HALAMAN PENGESAHAN.....</b>	<b>iii</b>
<b>SURAT PERNYATAAN .....</b>	<b>iv</b>
<b>HALAMAN PERSEMBAHAN .....</b>	<b>v</b>
<b>HALAMAN MOTTO .....</b>	<b>vi</b>
<b>ABSTRAK .....</b>	<b>vii</b>
<b>ABSTRACT .....</b>	<b>viii</b>
<b>RINGKASAN .....</b>	<b>ix</b>
<b>PRAKATA .....</b>	<b>x</b>
<b>PERNYATAAN PERSETUJUAN PUBLIKASI.....</b>	<b>xi</b>
<b>DAFTAR ISI.....</b>	<b>xii</b>
<b>DAFTAR TABLE .....</b>	<b>xiv</b>
<b>DAFTAR GAMBAR.....</b>	<b>xvi</b>
<b>DAFTAR KODE PROGRAM .....</b>	<b>xviii</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>xix</b>
<b>BAB 1 PENDAHULUAN .....</b>	<b>1</b>
1.1    Latar belakang.....	1
1.2    Rumusan Masalah .....	2
1.3    Batasan Masalah.....	2
1.4    Tujuan dan Manfaat .....	3
<b>BAB 2 TINJAUAN PUSTAKA.....</b>	<b>4</b>
2.1    Sistem Informasi.....	4
2.2    Voting .....	4
2.3    E-Voting .....	4
2.4    Sistem Informasi E-Voting .....	5

<b>2.5</b>	<b>OSIS .....</b>	6
<b>2.6</b>	<b>Sistem Voting Ketua OSIS SMAN 1 PAKUSARI JEMBER .....</b>	7
<b>2.7</b>	<b>Database Client Server .....</b>	8
<b>2.8</b>	<b>UML.....</b>	9
<b>2.9</b>	<b>Karya Tulis yang mendahului.....</b>	16
<b>2.10</b>	<b>State Of The Art .....</b>	18
<b>BAB 3. METODE KEGIATAN .....</b>		22
<b>3.1.</b>	<b>Waktu dan Tempat .....</b>	22
<b>3.2.</b>	<b>Alat dan Bahan .....</b>	22
<b>3.3.</b>	<b>Metode Kegiatan.....</b>	23
<b>3.4</b>	<b>Alur Pemilihan e-voting.....</b>	24
<b>3.5</b>	<b>Database Client Server E-Voting.....</b>	25
<b>3.6</b>	<b>Jadwal Kegiatan .....</b>	26
<b>BAB 4. HASIL DAN PEMBAHASAN .....</b>		27
<b>4.1</b>	<b>Communication .....</b>	27
<b>4.2</b>	<b>Quick Plan .....</b>	27
<b>4.3</b>	<b>Modeling Quick Design .....</b>	27
<b>4.4</b>	<b>Pembuatan Aplikasi .....</b>	45
<b>4.5</b>	<b>Flowcart Sistem Pemilihan E-voting Dan Manual.....</b>	98
<b>BAB 5. KESIMPULAN DAN SARAN .....</b>		100
<b>5.1</b>	<b>Kesimpulan .....</b>	100
<b>5.2</b>	<b>Saran.....</b>	100
<b>DAFTAR PUSTAKA .....</b>		101
<b>LAMPIRAN.....</b>		103

## DAFTAR TABLE

	Halaman
Tabel 2.1 Alur Pemilihan Ketua osis SMAN 1 PAKUSARI JEMBER .....	7
Tabel 2.2 simbol-simbol usecase diagram .....	10
Tabel 2.3 simbol-simbol class diagram.....	12
Tabel 2.4 simbol-simbol sequence diagram.....	14
Tabel 2.5 simbol-simbol statechart diagram .....	16
Tabel 2.6 Perbandingan Antara system informasi yang telah dikembangkan .....	18
Tabel 3.1 Skenario <i>IP Address E-voting</i> .....	26
Tabel 3.2 jadwal kegiatan yang dilakukan.....	26
Tabel 4. 1 Definisi Aktor .....	28
Tabel 4. 2 Definisi Usecase.....	29
Tabel 4. 3 Skenario Usecase Menambah pemilih .....	31
Tabel 4. 4 Skenario <i>usecase</i> mengubah pemilih .....	32
Tabel 4. 5 Skenario <i>usecase</i> menghapus pemilih.....	32
Tabel 4.6 Skenario <i>usecase</i> melihat pemilih.....	33
Tabel 4.7 Skenario <i>usecase</i> mencari pemilih.....	34
Tabel 4.8 Skenario <i>usecase</i> mencetak kode pemilih.....	34
Tabel 4. 9 Skenario Usecase Menambah kandidat.....	35
Tabel 4. 10 Skenario Usecase Mengubah kandidat .....	35
Tabel 4. 11 Skenario <i>usecase</i> menghapus kandidat .....	36
Tabel 4..12 Skenario <i>usecase</i> melihat kandidat .....	36
Tabel 4..13 Skenario <i>usecase</i> mencari kandidat .....	37
Tabel 4..14 Skenario <i>usecase</i> upload foto kandidat.....	38
Tabel 4. 15 Skenario Usecase Menambah user (panitia MPK) .....	38
Tabel 4. 16 Skenario Usecase Mengubah user (panitia MPK) .....	39
Tabel 4. 17 Skenario <i>usecase</i> menghapus user (panitia MPK).....	39
Tabel 4..18 Skenario <i>usecase</i> melihat user (panitia MPK) .....	40
Tabel 4.19 Skenario <i>usecase</i> mencari user (panitia MPK) .....	40

Tabel 4.20 Skenario <i>usecase</i> login user (panitia MPK) .....	41
Tabel 4.21 Skenario <i>usecase</i> logout user (panitia MPK) .....	42
Tabel 4.22 Skenario <i>usecase</i> Cetak rekapitulasi .....	42
Tabel 4.23 Skenario <i>usecase</i> Memasukkan kode pemilih .....	43
Tabel 4.24 Skenario <i>usecase</i> Memilih kandidat .....	44
Tabel 4.25 Skenario <i>usecase</i> Melihat kandidat.....	45
Tabel 4. 26 Tabel pemilih .....	47
Tabel 4. 27 Pengujian <i>black box</i> mengelola pemilih .....	51
Tabel 4. 28 Pengujian <i>black box</i> cetak kode pemilih.....	55
Tabel 4. 29 Tabel kandidat.....	58
Tabel 4. 30 Pengujian <i>black box</i> mengelola kandidat.....	64
Tabel 4. 31 Tabel USER .....	67
Tabel 4. 32 Pengujian <i>black box</i> mengelola user .....	72
Tabel 4. 33 Pengujian <i>black box</i> validasi user .....	82
Tabel 4. 34 Tabel kandidat.....	84
Tabel 4. 35 Pengujian black box cetak rekapitulasi.....	36
Tabel 4. 36 Pengujian black box login pemilih.....	91
Tabel 4. 37 Tabel kandidat.....	94
Tabel 4. 38 Pengujian black box memilih kandidat.....	96
Tabel 4. 39 Flowchart Sistem Pemilihan E-voting .....	98
Tabel 4. 40 Flowchart Sistem Pemilihan Manual .....	99

## DAFTAR GAMBAR

	Halaman
Gambar 2.1 contoh Use Case Diagram .....	10
Gambar 2.2 Contoh Class Diagram .....	12
Gambar 2.3 Contoh Sequence.....	14
Gambar 2.4 Contoh Statechart Diagram .....	15
Gambar 3.1 Paradigma Pembuatan <i>Prototype</i> .....	23
Gambar 3.2 alur pemilihan pemilih .....	24
Gambar 3.3 Alur Database Database Client Server Pada E-voting .....	25
Gambar 4. 1 <i>Usecase diagram</i> .....	28
Gambar 4. 2 <i>Class diagram</i> mengelola pemilih.....	46
Gambar 4. 3 Design <i>mockup</i> mengelola pemilih .....	47
Gambar 4.4 Form Pemilih.....	51
Gambar 4. 5 Design <i>mockup</i> cetak kode pemilih.....	53
Gambar 4. 6 <i>form</i> Cetak Kode Pemilih.....	55
Gambar 4. 7 <i>Class diagram</i> mengelola kandidat .....	57
Gambar 4. 8 Design <i>mockup</i> mengelola kandidat.....	57
Gambar 4. 9 Design <i>mockup</i> mencari calon kandidat dari data pemilih.....	58
Gambar 4. 10 <i>Script mengambil data kandidat dari data pemilih</i> .....	62
Gambar 4. 11 <i>form kandidat</i> .....	63
Gambar 4. 12 <i>form cari kandidat dari data pemilih</i> .....	63
Gambar 4. 13 <i>form cari foto</i> .....	63
Gambar 4. 14 <i>Class diagram</i> mengelola user .....	66
Gambar 4. 15 Design <i>mockup</i> mengelola user.....	67
Gambar 4. 16 <i>form mengelola user</i> .....	72
Gambar 4. 17 <i>Class diagram</i> login dan logout .....	76
Gambar 4. 18 <i>Class diagram</i> login .....	77
Gambar 4. 19 <i>Activity</i> menu sebelum <i>login</i> .....	79
Gambar 4. 20 <i>Activity</i> menu sesudah <i>login MPK (administrator)</i> .....	80

Gambar 4. 21 <i>Activity</i> menu sesudah <i>login MPK (panitia tps)</i> .....	80
Gambar 4. 22 <i>Activity</i> menu sesudah <i>login Staf Kesiswaan</i> .....	81
Gambar 4. 23 <i>Activity</i> menu sesudah <i>logout</i> .....	81
Gambar 4. 24 <i>Class diagram</i> cetak rekapitulasi .....	83
Gambar 4. 25 Design <i>mockup</i> cetak rekapitulasi suara.....	84
Gambar 4. 26 <i>form</i> cetak rekapitulasi suara.....	85
Gambar 4. 27 <i>Class diagram</i> login pemilih .....	87
Gambar 4. 28 Design <i>mockup</i> login pemilih.....	87
Gambar 4. 30 Design login pemilih .....	91
Gambar 4. 31 <i>Class diagram</i> kandidat.....	93
Gambar 4. 32 Design <i>mockup</i> pemilihan kandidat .....	94
Gambar 4. 33 <i>form</i> pemilihan kandidat.....	95

## **DAFTAR KODE PROGRAM**

	Halaman
Kode Program 4.1 model pemilih .....	47
Kode Program 4.2 form cetak kode pemilih .....	54
Kode Program 4.3 model kandidat.....	58
Kode Program 4.4 kode upload foto .....	61
Kode Program 4.5 kode mencari kandidat dari data pemilih.....	61
Kode Program 4.6 model user.....	67
Kode Program 4.7 kode login user.....	77
Kode Program 4.8 kode hak akses login user .....	78
Kode Program 4.9 kode cetak rekapitulasi hasil suara .....	84
Kode Program 4.10 kode model login pemilih .....	88
Kode Program 4.11 kode login pemilih .....	89
Kode Program 4.12 model pemilihan kandidat.....	94

## **DAFTAR LAMPIRAN**

	Halaman
Lampiran 1. Laporan Hasil Rekapitulasi Suara .....	104
Lampiran 2 Hasil Cetak Kode Login Pemilih.....	105

## **BAB 1 PENDAHULUAN**

### **1.1 Latar belakang**

Setiap sekolah memiliki ketentuan dan kebijakan masing-masing dalam menyejahterakan sekolahnya. Salah satunya OSIS yang sangat berperan penting bagi sekolah karena OSIS sebagai satuan atau kelompok kerjasama para siswa yang dibentuk dalam usaha mencapai tujuan bersama, oleh karena itu pemilihan ketua OSIS harus dilakukan secara benar , ketua OSIS yang diharapkan mampu membawa sekolah kearah yang lebih baik lagi, Pelaksanaan pemilihan harus berlangsung LUBER (Langsung Umum Bebas Rahasia). Hal ini dilakukan untuk melatih para siswa dalam menjalankan salah satu asas dalam demokrasi mulai dari pemberian kartu suara, pencoblosan, memasukan kartu suara, pencelupan atau penandaan jari serta perhitungan suara

Kemajuan teknologi yang ada saat ini terus berkembang, laju perkembangan teknologi yang sangat cepat memang tak bisa kita hindari dan hentikan, begitu juga dengan perangkat-perangkat elektronik, Hal ini semakin mempermudah manusia dalam melakukan suatu kegiatan, termasuk kegiatan pemilihan ketua OSIS, sehingga memberikan kemudahan bagi semua kalangan yang tidak bisa lepas dari teknologi, termasuk kalangan muda saat ini yang tak bisa lepas, sampai banyak berbagai tempat termasuk lembaga yang kini telah dilengkapi kecanggihan perangkat elektronik yang menggantikan sistem manual termasuk juga sekolah sekolah yang rata-rata sudah menggantikan sistem manual menggunakan kecanggihan teknologi yang dapat mempermudah proses (e-voting) pemilihan ketua OSIS

E-voting didefinisikan secara umum sebagai proses pemungutan suara yang memungkinkan pemilih untuk memberikan suara (aman dan rahasia) melalui internet atau intranet yang memungkinkan pemilih untuk memberikan suaranya di tempat yang mereka pilih yang saling terhubung. Pengertian lain e-voting adalah pemungutan suara yang dilakukan secara elektronik (digital) mulai dari proses

pendaftaran pemilih, pelaksanaan pemilihan, penghitungan suara, dan pengiriman hasil suara (Ali Rokhman, 2011).

Pemilihan Ketua OSIS di SMAN 1 PAKUSARI Jember masih dilakukan secara konvensional selain itu dalam pemilihan ketua OSIS mengeluarkan tenaga dan memerlukan waktu yang banyak dalam menjelaskan visi misi calon serta pengumpulan dan perhitungan suara, karena panitia dan calon ketua OSIS harus keliling ke tiap-tiap kelas mulai dari kelas 10 hingga kelas 12 dan ruang guru sehingga dapat mengganggu kegiatan belajar mengajar

Mengingat pentingnya pemilihan ketua OSIS, dengan perkembangan teknologi tersebut sekolah seharusnya menggunakan pelaksanaan pemungutan suara *e-voting* yang akan diterapkan dengan penggunaan komputer agar lebih efisien. Disini penulis membuat sebuah sistem pemilihan ketua OSIS berbasis dekstop, Dengan sistem tersebut tentunya akan sangat membantu sekolah dalam memilih ketua OSIS seperti kemudahan, efisiensi, peningkatan partisipasi pemilih, dan keakuratan penghitungan suara secara bijaksana dan menghindari kecurangan, sistem dari penulis sangat mudah, cepat, efisien dan akurat serta membuat pemilih percaya atas hasil pemilihan

## 1.2 Rumusan Masalah

Berkenaan dengan latar belakang diatas, maka rumusan masalah yang akan diambil adalah

- a. Bagaimana cara membuat pelaksanaan pemilihan tidak mengeluarkan tenaga dan memerlukan waktu yang banyak?
- b. Bagaimanacara membuat Membuat Sistem Informasi E-voting pemilihan ketua OSIS SMAN 1 Pakusari Jember Berbasis Dekstop?

## 1.3 Batasan Masalah

Permasalahan yang dibahas oleh penulis dalam pembuatan tugas akhir ini adalah sebagai berikut:

- a. Sistem informasi ini menggunakan database MY SQL dan Bahasa pemrograman java
- b. Sistem informasi e-voting ini kandidat calon ketua OSIS sudah ditentukan oleh panitia
- c. Sistem informasi e-voting ini mencakup pemilihan ketua OSIS di SMAN 1 Pakusari Jember

#### **1.4 Tujuan dan Manfaat**

Adapun tujuan yang ingin dicapai adalah

- a. Membuat cara pelaksanaan pemilihan tidak mengeluarkan tenaga dan memerlukan waktu yang banyak
- b. Membuat sistem informasi e-voting berbasis dekstop di SMAN 1 Pakusari Jember

Adapun beberapa manfaat yang diperoleh, yaitu:

- a. Bagi Mahasiswa
  - 1) Mengasah kemampuan dan menerapkan hasil pembelajaran yang diterima selama perkuliahan di Politeknik Negeri Jember
- b. Bagi Sekolah
  - 1) Membantu Bagaimana cara membuat pelaksanaan pemilihan tidak mengeluarkan tenaga dan memerlukan waktu yang banyak
  - 2) Membantu proses pemilihan ketua OSIS dengan hemat tenaga dan waktu
- c. Bagi Politeknik Negeri Jember
  - 1) Menjalin hubungan kerjasama dengan instansi lain
  - 2) Menjadi refrensi bagi mahasiswa untuk tugas akhir

## **BAB 2 TINJAUAN PUSTAKA**

### **2.1 Sistem Informasi**

Menurut O'Brian dikutip oleh Yakub (2012:17) pada buku Pengantar Sistem Informasi,sistem informasi (*information system*) merupakan kombinasi teratur dari orang-orang, perangkat keras, perangkat lunak, jaringan komunikasi, dan sumber daya data yang mengumpulkan, mengubah, menyebarkan informasi dalam sebuah organisasi.

Pengertian sistem informasi menurut John F. Nash Sistem Informasi adalah kombinasi dari manusia, fasilitas atau alat teknologi, media, prosedur dan pengendalian yang bermaksud menata jaringan komunikasi yang penting, proses atas transaksi-transaksi tertentu dan rutin, membantu manajemen dan pemakai intern dan ekstern dan menyediakan dasar pengambilan keputusan yang tepat.

### **2.2 Voting**

Menurut Darmawan, Ikhsan., Nurhanjati, Nurul., Kartini., Evida (2014). voting dapat diartikan sebagai tindakan menggunakan hak untuk memilih dalam sebuah kegiatan pemilihan umum. Mengutip Evans, voting berkaitan sebuah pilihan tentu saja, sebuah voting yang dimaksudkan tersebut selalu berkaitan dengan pilihan akan pemimpin yang menurut seorang pemilih termasuk yang paling mewakili mereka

### **2.3 E-Voting**

Menurut Darmawan, Ikhsan., Nurhanjati, Nurul., Kartini., Evida (2014). E-voting secara sederhana dapat diartikan sebagai penggunaan hak pilih dalam sebuah pemilu dengan menggunakan bantuan teknologi (secara elektronik).

Pada umumnya, cara memilih adalah dengan menggunakan cara mencoblos atau menandai di kertas surat suara. Akan tetapi, seiring perkembangan

teknologi, terdapat teknik lain yaitu electronic voting, electronic voting adalah suatu metode pengumpulan suara dengan menggunakan perangkat elektronik

Menurut Kim (2006) yang mendefinisikan *e-voting* sebagai pemungutan suara resmi yang aman dan rahasia mempergunakan mesin elektronis. Kim juga menjelaskan definisi *e-voting* yang lebih luas, yang diistilahkan sebagai *e-voting* dengan skala penuh (*full scale e-voting*) yang mencakup ruang lingkup dan alat bantu yang dipergunakan. Dalam hal ruang lingkup, *e-voting* didefinisikan sebagai aksi pemilihan dan aksi wacana serta artikulasi politis dalam ruang maya (*cyberspace*) dalam periode pemilihan. Sementara dalam hal alat bantu, *e-voting* mencakup pemilihan melalui internet, *personal digital assistance*, *mobile phones*, yang memungkinkan pemilih untuk memberikan suaranya di tempat yang mereka pilih, maupun pemilihan di tempat pemungutan suara yang dilengkapi dengan alat bantu pemungutan suara elektronis.

#### **2.4 Sistem Informasi E-Voting**

Menurut O'Brian dikutip oleh Yakub (2012:17) pada buku Pengantar Sistem Informasi, sistem informasi (*information system*) merupakan kombinasi teratur dari orang-orang, perangkat keras, perangkat lunak, jaringan komunikasi, dan sumber daya data yang mengumpulkan, mengubah, menyebarluaskan informasi dalam sebuah organisasi.

Menurut Darmawan, Ikhsan., Nurhanjati, Nurul., Kartini., Evida (2014). *e-voting* secara sederhana dapat diartikan sebagai penggunaan hak pilih dalam sebuah pemilihan dengan menggunakan bantuan teknologi (secara elektronik).

Sistem informasi *e-voting* dapat didefinisikan sebagai pelaksanaan pemungutan suara yang dilakukan secara elektronik (digital) mulai dari proses pendaftaran pemilih, pelaksanaan pemilihan, penghitungan suara, dan pengiriman hasil suara.

## 2.5 OSIS

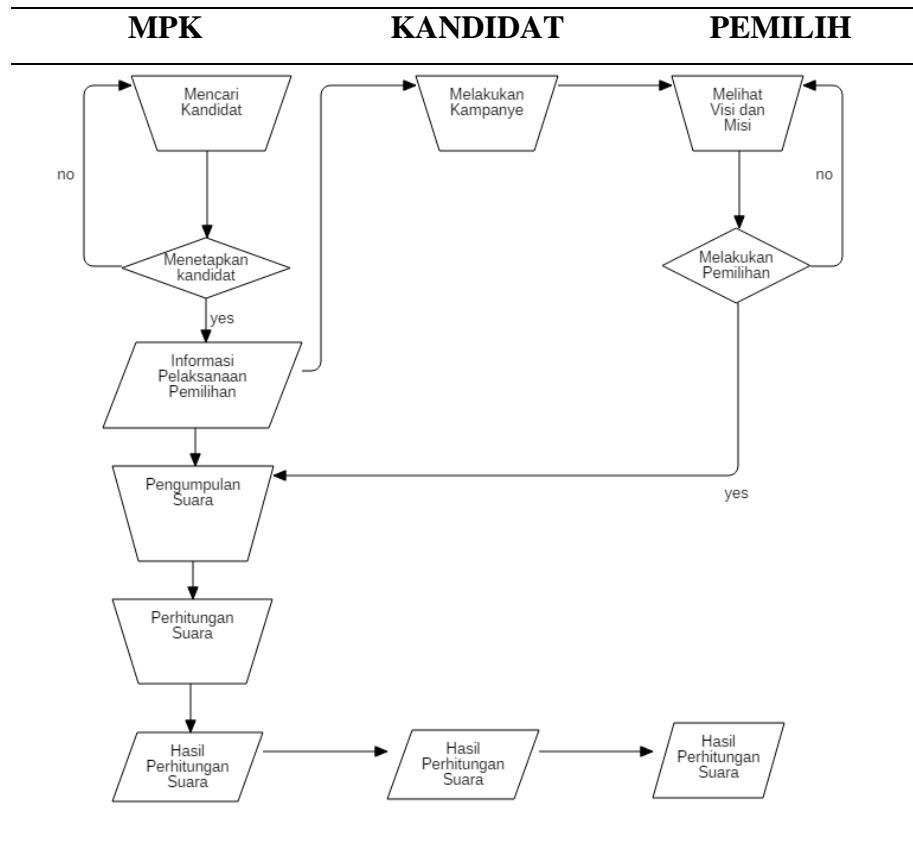
Menurut Syamsul, Z. (2014) OSIS (Organisasi Siswa Intra Sekolah) adalah suatu organisasi yang berada di tingkat sekolah di Indonesia yang dimulai dari sekolah menengah yaitu sekolah menengah pertama (SMP) dan sekolah menengah atas (SMA), OSIS diurus dan dikelola oleh murid-murid yang terpilih untuk menjadi pengurus OSIS, Biasanya organisasi ini memiliki seorang pembimbing seorang guru yang dipilih oleh pihak sekolah

Tujuan pendidikan dan pembinaan generasi muda yang ditetapkan baik di dalam pembukaan UUD 1945 maupun di dalam garis-garis besar, Haluan negara amat luas ligkupnya , maka diperlukan sekolah sebagai lingkungan pendidikan yang merupakan jalur pendidikan formal yang sangat penting dan strategis bagi upaya mewujudkan tujuan tersebut, baik melalui proses belajar mengajar maupun melalui kegiatan extrakurikuler

Anggota OSIS adalah seluruh siswa yang berada pada satu sekolah tempat OSIS itu berada,Seluruh anggota OSIS berhak untuk memilih calonnya untuk kemudian menjadi pengurus OSIS

## 2.6 Sistem Voting Ketua OSIS SMAN 1 PAKUSARI JEMBER

Tabel 2.1 Alur Pemilihan Ketua OSIS SMAN 1 PAKUSARI JEMBER



Sumber dokumentasi SMAN 1 Pakusari Jember

Berdasarkan tabel 2.1: MPK (majelis perwakilan kelas) melakukan penyeleksian kandidat ketua OSIS dilalui atas persetujuan kesepakatan langkah pertama MPK komisi A sebagai pengawas dan penanggung jawab kinerja kepengurusan OSIS melakukan seleksi dengan cara mencari anggota OSIS yang aktif dan cekatan menjadi 5 orang calon kandidat lalu melakukan seleksi lagi untuk menetapkan 3 orang kandidat melalui rapat untuk menentukan siapa saja kandidat ketua OSIS yang akan dipilih oleh semua warga sekolah setelah itu MPK memberitahu kapan pelaksanaan pemilihan dilakukan. Keesokan harinya para calon ketua OSIS melakukan kampanye dengan membuat *pamphlet/benner* ditembok ataupun mading sekolah yang berisikan visi, misi dan prestasi masing-masing calon dengan menggunakan dana pribadi selama 3 hari sebelum pemilihan, pada saat

pemilihan panitia yang bertugas mengumpulkan suara berjumlah 4 orang yaitu MPK, atas pengawasan langsung dari waka kesiswaan

Pada waktu pemilihan panitia dan calon ketua OSIS keliling ke tiap-tiap kelas mulai dari kelas 10 hingga kelas 12 dan ruang guru, MPK memberikan intruksi untuk masuk ke kelas masing-masing dan memberikan penjelasan cara memilih calon dan menuliskan nomer urut di papan tulis sebagai nomer urut masing masing calon kemudian para calon menjelaskan visi dan misinya secara bergantian lalu panitia (MPK) memberikan kertas kecil bersetempel ke masing masing siswa sebagai surat suara yang nantinya di masukkan ke kotak suara bagi siswa yang tidak masuk sekolah hak suara dianggap hangus, ATK yang digunakan pada saat pemilihan adalah ketras, spidol dan kardus sebagai kotak suara waktu pengumpulan suara memakan waktu sekitar 60 menit, cara perhitungan suara dihitung setelah melakukan pemungutan suara yang dilakukan secara terbuka yang disaksikan oleh perwakilan OSIS ,MPK, guru dan siswa yang membutuhkan waktu 15 menit suara terbanyak akan otomatis menjadi ketua OSIS dan yang kedua menjadi wakil dan ketiga otomatis menjadi bendahara dan jika hasil imbang maka dilakukan test visi dan misi masing masing kandidat di hadapan guru-guru, setelah selesai melakukan pemilihan Staf kesiswaan melaporkan hasil pemilihan kepada Waka kesiswaan setelah dua minggu baru dilakukan pelantikan yang dilakukan oleh kepala sekolah, Hasil keputusan akhir murni hasil jumlah suara dari siswa dan guru dan dana yang diperlukan saat pemilihan ketua OSIS kurang lebih sebesar 500.000 termasuk benner,kertas suara dan konsumsi panitia

## **2.7 Database Client Server**

Menurut Fathansyah (2015), Sebagaimana Sistem Tersentralisasi, arsitektur ketiga ini juga diterapkan pada sebuah sistem jaringan. Sistem Client-Server ini ditujukan untuk mengatasi kelemahan-kelemahan yang terdapat pada sistem tersentralisasi sebelumnya (baik sentralisasi pada aplikasi dan basis data maupun sentralisasi hanya pada basis data). Kelemahan pada bentuk sistem tersentralisasi yang pertama, yaitu beratnya beban server yang harus menangani semua proses, diatasi dengan membagi beban itu menjadi 2 bagian: client (yang

menjalankan aplikasi basis data) dan server (yang menjalankan DBMS dan berisi basis data) pada mesin yang berbeda. Sedang kelemahan pada bentuk sistem tersentralisasi yang kedua, yaitu padatnya lalu lintas data Antara server dan work-station diatasi dengan mekanisme transfer data yang lebih efisien.

Sistem ini terdiri atas dua komponen (mesin) utama, yaitu client dan server, client berisi aplikasi basis data dan server berisi DBMS dan basis data. Setiap aktivitas yang dikehendaki para pemakai akan lebih dulu ditangani oleh client. Client selanjutnya mengupayakan agar semua proses ‘sebisa mungkin’ ditangani sendiri. Jika ada proses yang harus melibatkan data yang tersimpan pada basis data, barulah client melakukan ‘kontak’ dengan server, katakanlah ada kebutuhan untuk mendapatkan baris-baris data (row) dengan kriteria tertentu (proses query) dari tabel data bernama T. Pada bentuk Sistem Tersentralisasi yang kedua, kebutuhan ini dijawab dengan mentransfer lebih dulu semua baris data (row) yang ada di tabel T tersebut dari server ke work-station, barulah kemudian di work-station dilakukan pemilihan baris data (query) terhadap tabel T tersebut. Sementara pada sistem client-server, untuk memenuhi kebutuhan itu client akan mengirimkan message (perintah) query pengambilan data. Selanjutnya, server yang menerima message tersebut akan menjalankan query tersebut (melakukan pencarian baris data terhadap table T) dan hanya hasilnya (jadi tidak seluruh isi tabel T) yang akan dikirimkan kembali ke client. Dengan begitu, Transfer datanya menjadi jauh lebih efisien

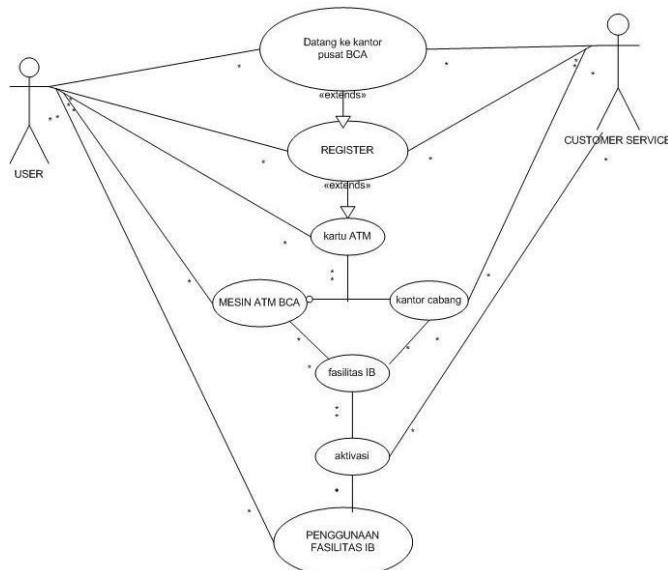
## 2.8 UML

Menurut S,Rosa A dan M. Salahuddin (2016:9) Pada perkembangan teknik pemrograman berorientasi objek,muncullah sebuah standarisasi Bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML).UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan,membangun dan dokumentasi dari sistem perangkat lunak. UML merupakan Bahasa visual untuk pemodelan dan

komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung

### 2.8.1 Use Case Diagram

Menurut S, Rosa A dan M. Salahuddin (2016:9) mengungkapkan bahwa “Use case atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat.”

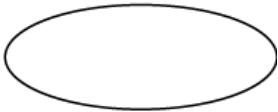
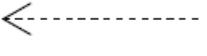


Gambar 2.1 contoh Use Case Diagram pendaftaran internet banking BCA

Tabel 2.2 simbol-simbol usecase diagram (Rosa A.S-M.Shalahuddin,2016)

Symbol	Nama simbol	Keterangan
	aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem infomasi yang akan dibuat sendiri, jadi walaupun symbol dari actor adalah gambar orang, tapi actor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor

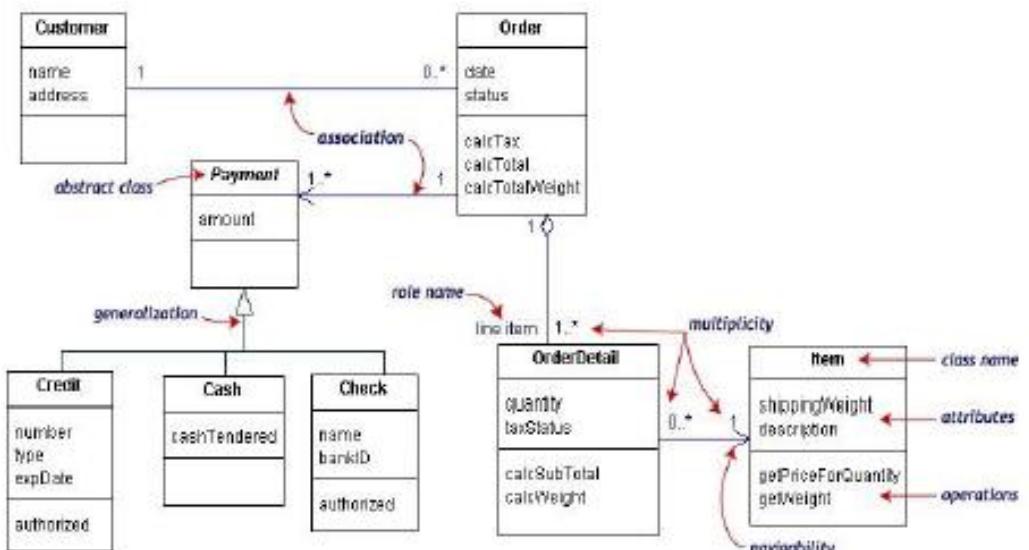
Tabel 2.2 simbol-simbol usecase diagram (Rosa A.S-M.Shalahuddin,2016)  
(Lanjutan)

Symbol	Nama simbol	Keterangan
	Use case	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor, biasanya dinyatakan dengan menggunakan kata kerja awal frase nama use case
	Include dan uses	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan case ini. Include berarti use case yang ditambahkan akan selalu dipanggil saat use case tambahan di jalankan
	Asosiasi/associaton	Komunikasi Antara actor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor
	generalisasi	Hubungan generalisasi dan spesialisasi (umum-khusus) Antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya

### 2.8.2 Class Diagram

Menurut S.Rosa A dan M. Salahuddin (2016:9), mengungkapkan bahwa "Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefenisian kelas-kelas yang akan dibuat untuk membangun sistem". Berikut contoh class diagram

Diagram kelas dibuat agar pembuat program atau programmer membuat kelas kelas sesuai rancangan di dalam diagram kelas agar Antara dokumentasi perancangan dan perangkat lunak sinkron.

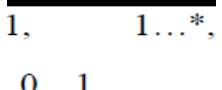
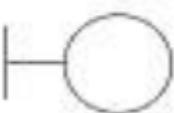


Gambar 2.2 Contoh Class Diagram transaksi pembelian barang

Tabel 2.3 simbol-simbol class diagram (Rosa A.S-M.Shalahuddin,2016)

Symbol	Nama symbol	Deskripsi
	class	Menggambarkan sesuatu yang mengkapsulkan informasi di class menampung nama class, atribut dan method
	Package	Digunakan untuk mengelompokan kelas-kelas yang mempunyai persamaan

Tabel 2.3 simbol-simbol class diagram (Rosa A.S-M.Shalahuddin,2016)  
(Lanjutan)

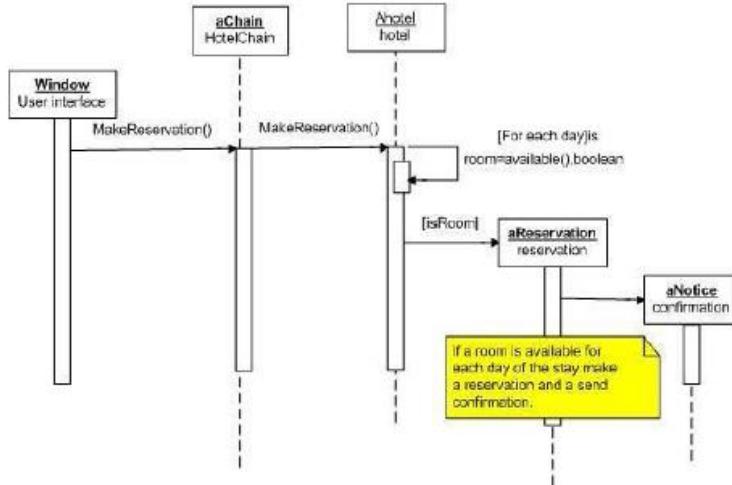
Symbol	Nama symbol	Deskripsi
	Asosiasi	Asosiasi yang menghubungkan class dengan class multiplycity
	Boundary class	Menggambarkan class yang menjadi antar muka actor dengan sistem
 Nama_interface	Antarmuka / interface	Sama dengan konsep interface dalam pemrograman berorientasi objek
	Aggregation	Menggambarkan suatu class terdiri dari class lain atau suatu class adalah bagian dari class lain
	Generalization	Merupakan sebuah taxonomic relationship Antara class yang lebih umum dengan class yang lebih khusus

### 2.8.3 Sequence Diagram

Menurut S,Rosa A dan M. Salahuddin (2016:9), Diagram sekuen (*sequence diagram*) menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterimaantara objek. Oleh karena itu untuk menggambar diagram sekuen, maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki keas yang diinstansiasi menjadi objek itu.

Banyak diagram sekuen yang harus digambar adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri, atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada

diagram sekuen, sehingga semakin banyak use case yang didefinisikan, maka diagram sekuen yang harus dibuat juga semakin banyak.



Gambar 2.3 Contoh Sequence Diagram Pemesanan Kamar Di Hotel

Tabel 2.4 simbol-simbol sequence diagram (Rosa A.S-M.Shalahuddin,2016)

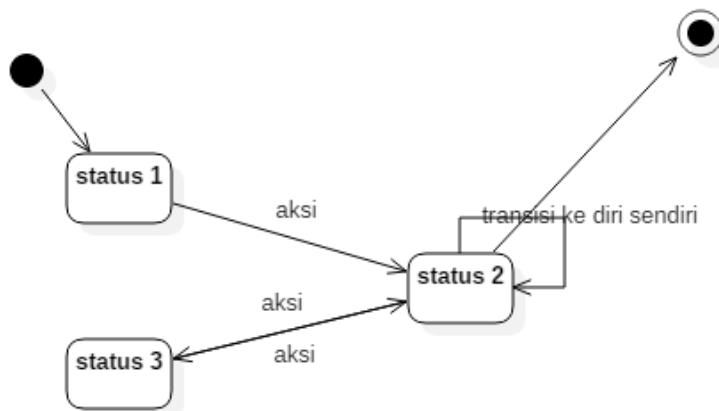
Symbol	Nama simbol	Keterangan
	Object lifi line / garis hidup	Menyatakan objek yang berinteraksi pesan dan menyatakan kehidupan suatu objek
	Action / waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan
	Pesan tipe sent	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarahkan pada objek yang dikirim

Tabel 2.4 simbol-simbol sequence diagram (Rosa A.S-M.Shalahuddin,2016)  
(Lanjutan)

Symbol	Nama simbol	Keterangan
→	Message (call)	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri
←	Message (return)	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah apabila mengarah pada objek yang menerima kembalian

#### 2.8.4 Statechart Diagram

Menurut S,Rosa A dan M. Salahuddin (2016:9), statechart diagram atau dalam Bahasa Indonesia disebut diagram mesin status atau sering juga disebut diagram status digunakan untuk menggambarkan perubahan status atau transisi status dari sebuah mesin atau sistem atau objek



Gambar 2.4 Contoh Statechart Diagram

Tabel 2.5 simbol-simbol statechart diagram (Rosa A.S-M.Shalahuddin,2016)

Symbol	Nama symbol	Keterangan
	Start / status awal	State atau initial state adalah state atau keadaan awal pada saat sistem mulai hidup
	End / status akhir	End atau final state adalah state keadaan akhir dari dasar hidup suatu sistem
	Event	State atau status adalah keadaan
	State	Sistem pada waktu tertentu, state dapat berubah jika ada event tertentu yang meniru perubahan tersebut

## 2.9 Karya Tulis yang mendahului

2.9.1 Rancang Bangun Sistem Informasi E-Voting Berbasis SMS (*Developing E-Voting Information System SMS Based*) (Adhi,Rizqi Andhestria , Harjono, *Universitas Muhammadiyah Purwokerto 2014*)

Voting adalah kegiatan yang sangat menentukan pada setiap perhelatan pemilihan,persoalan yang menjadi fokus perhatian bagi panitia penyelenggara pemilihan adalah bagaimana proses pemungutan suara dapat menjamin azas langsung, umum, bebas dan rahasia serta bagaimana hasil penghitungan suara dapat berlangsung jujur, transparan, dan dapat diakses oleh publik.

Salah satu kegiatan yang belum tersentuh teknologi yaitu proses pemilihan Ketua Organisasi Siswa Intra Sekolah (OSIS),di sekolah-sekolah semua prosesnya kebanyakan masih dilakukan secara manual, hal ini dirasa kurang efektif dengan berkembangnya teknologi sekarang ini.

Untuk memenuhi hal tersebut maka dibangun sebuah aplikasi E-voting berbasis SMS Gateway. Sistem ini dibangun menggunakan software Gammu dan dengan menggunakan Bahasa pemrograman PHP. Sistem ini menggunakan SMS

untuk melaksanakan voting, dalam hal ini siswa dapat memilih dan mengetahui hasil voting dengan mengirimkan SMS saja.

Untuk melakukan proses voting berbasis SMS Gateway ini pertama admin memasukan data siswa beserta no telepon siswa, kemudian memasukan data calon yang akan dipilih termasuk jenis dan kode pemilihan. Setelah itu admin memberitahukan kepada pemilih yaitu siswa melalui sms yang berisi nama, no urut calon ketua,kode pemilihan dan cara memilih dengan format sms yang benar.Format sms dalam perancangan ini yaitu untuk melakukan pemilihan digunakan format sms Kode Pemilihan#No Urut sedangkan untuk mengetahui hasil pemilihan digunakan format sms Hasil#Kode,

Diharapkan akan memberikan hasil voting yang akurat, cepat dan terpercaya serta bisa mengurangi angka siswa yang tidak memilih dan tidak akan mengganggu proses belajar mengajar.

#### 2.9.2 Sistem Informasi Pemilihan Ketua Umum Deperma Di Politeknik Negri Jember Berbasis Web (Nugraha, Dhany Eka, Politeknik Negri Jember 2015)

Di dunia teknologi yang semakin maju sekarang ini, manusia tidak bisa lepas dengan dunia internet,pembuatan sistem informasi Pemilihan Ketua Umum Deperma di Politeknik Negeri Jember Berbasis Web ini bertujuan untuk mempermudah bagi mahasiswa memilih calon ketua Deperma disana di karenakan era sekarang pemilihan dengan cara online sangat dibutuhkan .

Dalam penelitian Sistem Informasi Pemilihan Ketua Umum Deperma Di Politeknik Negri Jember Berbasis Web, karena Politeknik Negri Jember melakukan pemilihan ketua umum DEPERMA yang masih dilakukan secara manual. Pemilihan manual pun memiliki masalah seperti lambatnya penghitungan suara yang dilakukan kepanitiaan 2014 DEPERMA dalam menentukan jumlah suara. Dari permasalahan tersebut, maka perlu adanya sebuah aplikasi yang dibangun untuk mengelola laporan suara pemilihan yang telah dihasilkan oleh setiap mahasiswa yang memilih di Politeknik Negri Jember tersebut dapat diakses oleh para mahasiswa yang berupa tampilan website yang dapat didokumentasikan

menggunakan database dan menampilkan persentase suara hasil pemilihan secara online

Untuk melakukan proses voting berbasis web ini Admin mendaftarkan mahasiswa dengan mengisi data-data mahasiswa, kemudian admin memberikan password kepada mahasiswa untuk melakukan login pemilihan, didalam desain halaman pemilihan, mahasiswa dapat melihat profil setiap calon ketua yang akan dipilih dengan menekan *button* vote untuk memilih, setelah melakukan voting admin dan mahasiswa bisa melihat hasil voting. Sehingga para mahasiswa dapat menggunakan aplikasi web tersebut dengan secara tepat dan sesuai dengan apa yang dibutuhkan oleh para mahasiswa di Politeknik Negeri Jember tersebut

## 2.10 State Of The Art

Berdasarkan isi dari kedua karya tulis di atas maka tugas akhir yang berjudul “sistem informasi E-voting: studi kasus di SMAN 1 Pakusari Jember ini memiliki perbandingan sebagai berikut:

Tabel 2.6 Perbandingan Antara sistem informasi yang telah dikembangkan terlebih dahulu dengan sistem informasi yang dibuat oleh penulis

Aspek		Penulis		
No	Analisa	Aditya Dwi Kurniawan	Rizqi Andhestria Adhi, Harjono	Dhany Eka Nugraha
1	Judul	SISTEM INFORMASI E-VOTING KETUA OSIS SMAN 1 PAKUSARI JEMBER	Rancang Bangun Sistem Informasi E- Voting Berbasis SMS <i>(Developing E-Voting Information System SMS Based)</i>	Sistem Informasi Pemilihan Ketua Umum Deperma Di Politeknik Negri Jember Berbasis Web
2	Topik	Sistem Informasi	Sistem Informasi	Sistem Informasi

Tabel 2.6 Perbandingan Antara sistem informasi yang telah dikembangkan terlebih dahulu dengan sistem informasi yang dibuat oleh penulis (lanjutan)

Aspek		Penulis	
No	Analisa	Aditya Dwi Kurniawan	Rizqi Andhestria Adhi, Harjono
3	Objek	SMAN 1 Pakusari Jember	SMKN 1 Kaligondang
4	Bahasa Pemograman	Java,MySQL	PHP,MySQL
5	Metode	Prototype	Pengembangan (development research)
6	Tahun	2017	2014
7	Fitur	-Login Admin menambah,menghapus, mengedit data calon kandidat / pemilih -Login Panitia menambah pemilih, mencari,data pemilih -Login pemilih Menggunakan Kartu id yang di scan rfid reder untuk masuk ke halaman pemilihan -cetak kertas Audit bagi proses pemilihan	-login Admin -menambah, menghapus ,mengedit calon -menu menjalankan sms -hasil pemilihan - user menerima sms Yang berisi calon dan format untuk memilih dan mengecek hasil pemilihan -pemilih melakikan -

Tabel 2.6 Perbandingan Antara sistem informasi yang telah dikembangkan terlebih dahulu dengan sistem informasi yang dibuat oleh penulis (lanjutan)

Aspek		Penulis		
No	Analisa	Aditya Dwi	Rizqi Andhestria	Dhany Eka
		Kurniawan	Adhi, Harjono	Nugraha
7	Fitur	pemilih sebagai perhitungan secara manual jika dibutuhkan	-Melihat Hasil dengan mengirimkan format yang telah didapatkan	-Hasil pemilihan berupa rekapitulasi pemilihan

Berdasarkan table 2.4, maka hasil penelitian yang dilakukan penulis dan peneliti yang terdahulu terdapat beberapa persamaan dan perbedaan sebagai berikut:

a. Topik dan Objek

Topik yang dibahas pada ketiga karya tulis tersebut sama, yaitu Sistem Informasi, dan objeknya sama di bagian Pemilihan ketua organisasi

b. Metode

Metode yang digunakan pada Sistem Informasi E-voting Ketua OSIS SMAN 1 Pakusari Jember adalah *prototype*, pada Rancang Bangun Sistem Informasi E-Voting Berbasis SMS (*Developing E-Voting Information System SMS Based*) Pengembangan (development research) dan pada Sistem Informasi Pemilihan Ketua Umum Deperma Di Politeknik Negeri Jember Berbasis Web menggunakan metode waterfall. Penulis memilih menggunakan metode prototype adalah karena dapat menjalin komunikasi baik antara user dan tidak memerlukan waktu yang banyak

c. Perbedaan

Kelebihan dari karya tulis Sistem Informasi E-voting SMAN 1 Pakusari Jember yaitu adanya kertas audit yang berisi pilihan pemilih sebagai keamanan jika ada kecurangan hasil suara ,maka akan dihitung secara manual melalui kertas audit

## **BAB 3. METODE KEGIATAN**

### **3.1. Waktu dan Tempat**

Penelitian tugas akhir berjudul “Sistem Informasi E-Voting Pemilihan Ketua OSIS SMAN 1 Pakusari Jember” dilaksanakan pada Januari 2017 sampai dengan Maret 2017 bertempat di Politeknik Negeri Jember.

### **3.2. Alat dan Bahan**

#### **3.2.1 Bahan**

Bahan bahan yang dibutuhkan dalam proyek tugas akhir berjudul “Sistem Informasi E-Voting Pemilihan Ketua OSIS SMAN 1 Pakusari Jember ” adalah :

- 1) Data petugas antara lain: jumlah petugas, nama petugas, waktu.
- 2) Data pemilihan antara lain :pemilihan,pengumpulan suara, perhitungan suara,hasil pemilihan

#### **3.2.2 Alat**

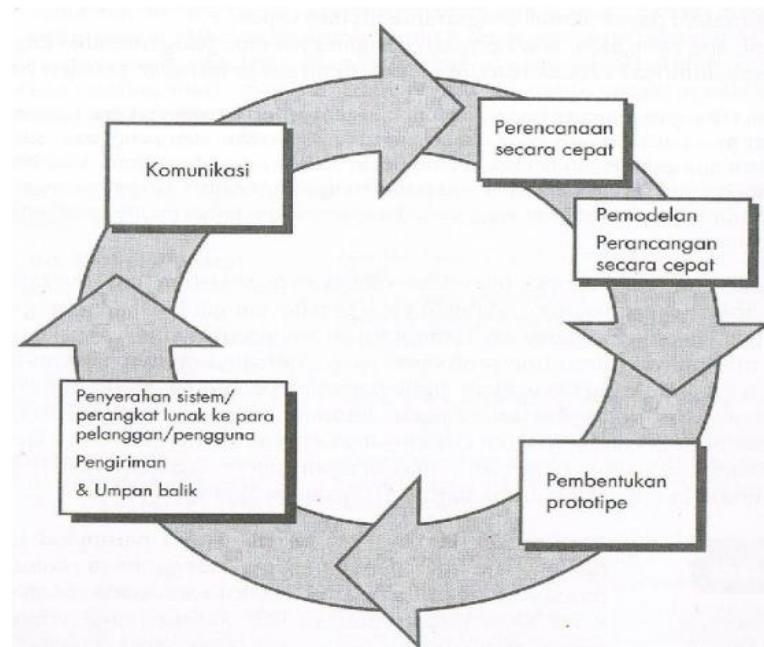
Adapun alat-alat yang digunakan dalam pembuatan desain program pembuatan perangkat lunak untuk mempermudah pembuatan Sistem Informasi E-Voting Pemilihan Ketua OSIS SMAN 1 Pakusari ini adalah terdiri dari perangkat keras dan perangkat lunak.

- a. Perangkat Keras yang digunakan adalah laptop dengan spesifikasi sebagai berikut:
  - 1) Processor Intel Core i3 3217U- 1.80 GHz, 3MB Cache
  - 2) RAM 2 Gb
  - 3) Hardisk 500GB
- b. Perangkat Lunak
  - 1) NetBeans IDE 8.2
  - 2) MySQL 5.7
  - 3) Sistem Operasi Windows 10

#### 4) Xampp 5.6.24-1

### 3.3. Metode Kegiatan

Metode yang digunakan pada tugas akhir yang berjudul “Sistem Informasi E-Voting Pemilihan Ketua OSIS SMAN 1 Pakusari Jember” adalah Metode *Prototype*. Adapun tahapan metode kegiatan ini dapat digambarkan pada gambar di bawah ini:



Gambar 3.1 Paradigma Pembuatan *Prototype* (Pressman, 2012:50)

Penjelasan fase-fase kegiatan ini adalah sebagai berikut :

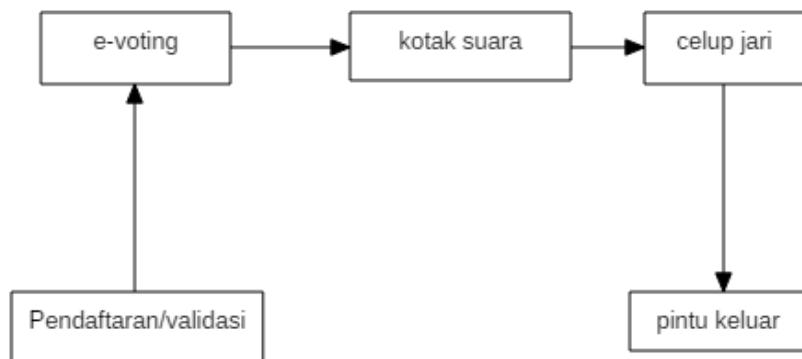
Dimulai dengan pengumpulan kebutuhan dan perbaikan, desain cepat, pembentukan prototipe, evaluasi pemilih terhadap prototipe, perbaikan prototipe dan produk akhir.

#### Tahapan-tahapan Model Prototyping

1. Komunikasi dan pengumpulan data awal, yaitu analisis terhadap kebutuhan pengguna. Meliputi kebutuhan pada saat pemilihan, pengumpulan suara dan perhitungan suara
2. Quick design (desain cepat), yaitu pembuatan desain secara umum untuk selanjutnya dikembangkan kembali.

3. Pembentukan prototype, yaitu pembuatan perangkat prototype, termasuk pengujian dan penyempurnaan.
4. Evaluasi terhadap *prototype*, yaitu mengevaluasi *prototypedan* memperhalus analisis terhadap kebutuhan pengguna.
5. Perbaikan prototype, yaitu pembuatan tipe yang sebenarnya berdasarkan hasil dari evaluasi prototype.
6. Produksi akhir, yaitu memproduksi perangkat secara benar sehingga dapat digunakan oleh pengguna.

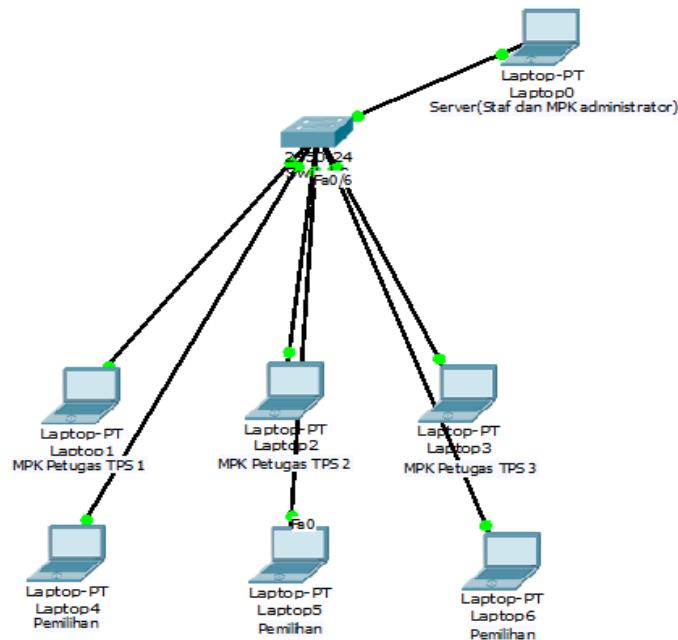
### 3.4 Alur Pemilihan e-voting



Gambar 3.2 alur pemilihan pemilih

Berdasarkan gambar 3.2: langkah-langkah pemilihan E-voting bagi pemilih prosesnya yaitu pertama pemilih datang ke TPS yang telah ditentukan, pemilih bisa bebas memilih kandidat kapan saja dan di TPS mana saja sebelum waktu yang ditentukan, saat ingin memilih pemilih wajib menunjukkan kartu pelajar sebagai syarat memilih ke panitia, setelah panitia menemukan data pemilih, panitia memberikan kode login yang nanti akan digunakan oleh pemilih di tempat e-voting, yang berfungsi untuk masuk ke halaman memilih kandidat, setelah itu pilihan pemilih akan otomatis disimpan ke database dan dicetak untuk selanjutnya dimasukan ke kotak suara yang nantinya akan juga dilakukan perhitungan suara secara manual jika ada permintaan dari pemilih atau protes, setelah itu pemilih mencelupkan jari sebagai bukti telah memilih

### 3.5 Database Client Server E-Voting



Gambar 3.3 Alur Database Database Client Server Pada E-voting

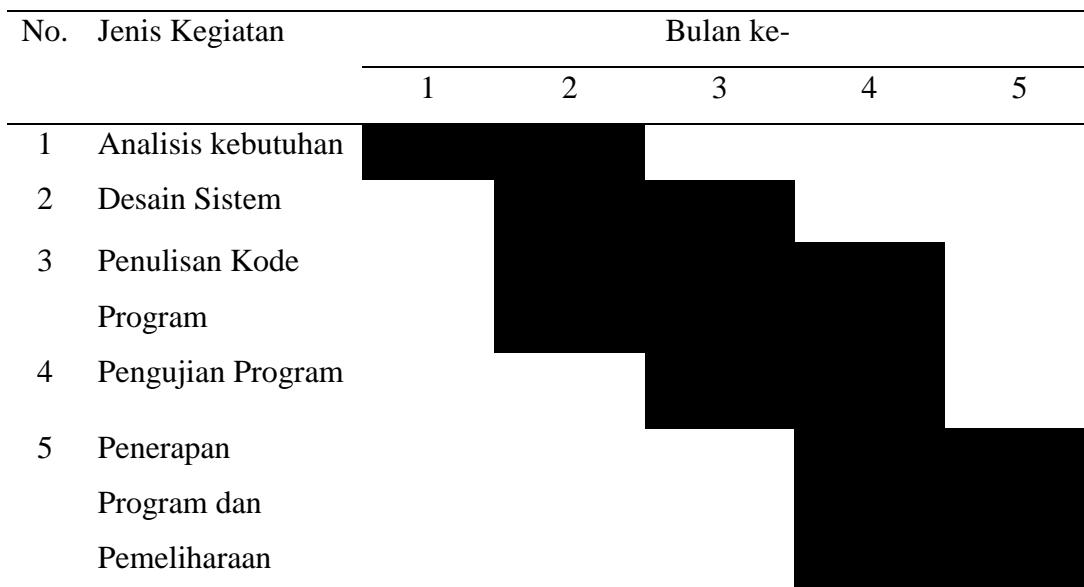
Berdasarkan gambar 3.3: pengaplikasian database client server di e-voting, dimana terdapat 3 TPS dan disetiap 1 TPS berisi 3 laptop yang telah dihubungkan menggunakan jaringan LAN yang nantinya digunakan oleh pemilih untuk memilih kandidat. Pemilih atau siswa bisa bebas ingin memilih di TPS mana saja sebelum waktu yang ditentukan habis, berikut gambaran pembagian ip adres, IP Address yang digunakan adalah type C dan Subnet Mask 255.255.255.0

Tabel 3.1 Skenario *IP Address E-voting*

Laptop	Berfungsi Sebagai	IP Address
Laptop0	Server (STAF dan MPK administrator)	192.168.33.1/24
Laptop1	Client (MPK Petugas TPS)	192.168.33.2/24
Laptop2	Client (MPK Petugas TPS)	192.168.33.3/24
Laptop3	Client (MPK Petugas TPS)	192.168.33.4/24
Laptop4	Client (Pemilih)	192.168.33.5/24
Laptop5	Client (Pemilih)	192.168.33.6/24
Laptop6	Client (Pemilih)	192.168.33.7/24

### 3.6 Jadwal Kegiatan

Tabel 3.2 jadwal kegiatan yang dilakukan



## BAB 4. HASIL DAN PEMBAHASAN

### 4.1 *Communication*

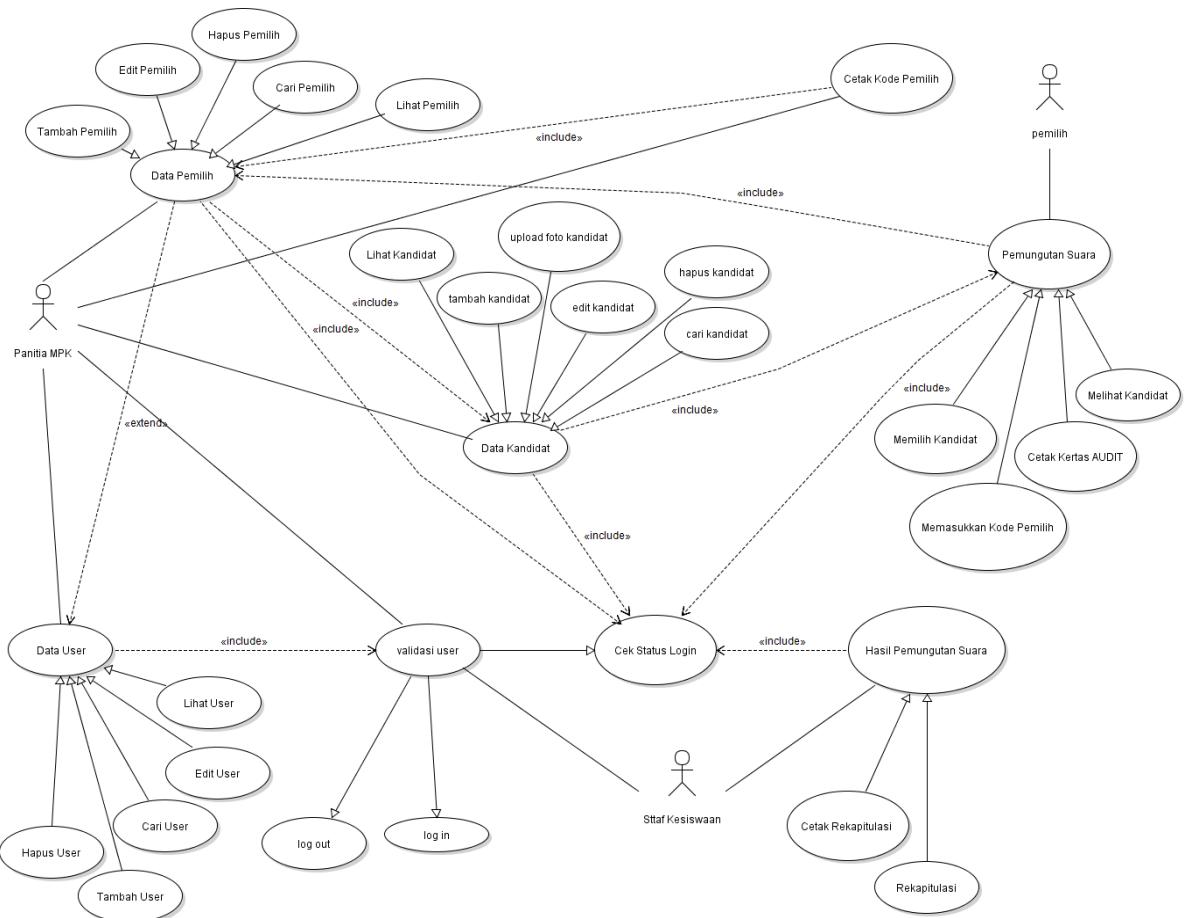
Tahap ini adalah tahap awal yang dilakukan dalam membangun Sistem Informasi E-voting Ketua OSIS SMAN 1 PAKUSARI JEMBER. Pada tahap ini dilakukan survei dari beberapa siswa dan panitia pemilihan di SMAN 1 PAKUSARI JEMBER. Setelah mengamati sistem pemilihan maka didapat beberapa fitur yang nantinya akan membantu sistem pemilihan ketua OSIS di SMAN 1 PAKUSARI

### 4.2 *Quick Plan*

Tahap selanjutnya adalah tahap Quick Plan yaitu tahap dimana developer mulai memikirkan perencanaan alur proses pembangunan prototype secara cepat. Developer akan menentukan dan menjelaskan input dan output dari prototype yang akan dibangun. Pada tahap ini developer akan menyesuaikan apakah analisanya sudah sama dengan apa yang diinginkan oleh pelanggan. Tahap ini juga merupakan tahap lanjutan dari *Communication*, karena di tahap ini semua data harus sudah lengkap untuk menentukan input dan outputnya secara benar agar dapat dilanjutkan ke tahap desain.

### 4.3 *Modeling Quick Design*

Tahap Modeling Quick Design yaitu membuat model untuk mempresentasikan perangkat lunak yang dapat dipahami oleh pengguna, sehingga pengguna mengerti bagaimana alur kerja perangkat lunak yang akan dibuat. Pengguna dapat melihat rancangan alur kerja sistem yang akan dimodelkan menggunakan permodelan UML. Untuk mengetahui jalannya proses didalam sistem informasi dapat di lihat pada Gambar 4.1 di bawah ini.



Gambar 4. 1 Use case diagram

#### 4.3.1 Definisi Aktor

Pada tabel 4.1 merupakan pendefinisian aktor kepada *usecase* sistem yang dikembangkan oleh pengembang :

Tabel 4. 1 Definisi Aktor

NO	Aktor	Deskripsi
1	Panitia MPK	Orang yang bertugas dan memiliki hak akses untuk mengolah data pemilih, data kandidat, data user, melihat data pemilih ,melihat data kandidat dan melihat data user
2	Staf Kesiswaan	Orang yang bertugas mencetak hasil rekapitulasi suara
3	Pemilih	Orang yang menggunakan sistem untuk memilih ketua OSIS

#### 4.3.2 Definisi *Usecase*

Pada tabel 4.2 ini merupakan pendefinisian *usecase* pada sistem informasi pemilihan ketua OSIS :

Tabel 4. 2 Definisi Usecase

<b>NO</b>	<b>Use Case</b>	<b>Deskripsi</b>
1	Menambah pemilih	Merupakan suatu proses untuk menambah data pemilih yang hanya dapat dilakukan oleh Panitia MPK
2	Mengubah pemilih	Merupakan suatu proses untuk mengubah data pemilih yang hanya dapat dilakukan oleh Panitia MPK
3	Menghapus pemilih	Merupakan suatu proses untuk menghapus data pemilih yang hanya dapat dilakukan oleh Panitia MPK
4	Melihat pemilih	Merupakan suatu proses untuk melihat data pemilih yang hanya dapat dilakukan oleh Panitia MPK
5	Mencari pemilih	Merupakan suatu proses untuk mencari data pemilih yang hanya dapat dilakukan oleh Panitia MPK
6	Mencetak Kode Pemilih	Merupakan suatu proses untuk mencetak kode pemilih yang hanya dapat dilakukan oleh Panitia MPK
7	Menambah kandidat	Merupakan suatu proses untuk menambah data kandidat yang hanya dapat dilakukan oleh Panitia MPK
8	Mengubah kandidat	Merupakan suatu proses untuk mengubah data kandidat yang hanya dapat dilakukan oleh Panitia MPK

Tabel 4. 2 Definisi Usecase (Lanjutan)

<b>NO</b>	<b>Use Case</b>	<b>Deskripsi</b>
9	Menghapus kandidat	Merupakan suatu proses untuk menghapus data kandidat yang hanya dapat dilakukan oleh Panitia MPK
10	Melihat kandidat	Merupakan suatu proses untuk melihat data kandidat yang ada pada database
11	Mencari kandidat	Merupakan suatu proses untuk mencari data kandidat yang hanya dapat dilakukan oleh Panitia MPK
12	Upload foto kandidat	Merupakan suatu proses untuk upload foto pada data kandidat yang hanya dapat dilakukan oleh Panitia MPK
13	Menambah user	Merupakan suatu proses untuk menambah data user yang hanya dapat dilakukan oleh Panitia MPK
14	Mengubah user	Merupakan suatu proses untuk mengubah data user yang hanya dapat dilakukan oleh Panitia MPK
15	Menghapus user	Merupakan suatu proses untuk menghapus data user yang hanya dapat dilakukan oleh Panitia MPK
16	Melihat user	Merupakan suatu proses untuk melihat data user yang hanya dapat dilakukan oleh Panitia MPK
17	Mencari user	Merupakan suatu proses untuk mencari data user yang hanya dapat dilakukan oleh Panitia MPK
18	Login	Merupakan suatu proses untuk user agar dapat masuk kedalam sistem dan dapat menambah, mengubah, menghapus data yang ada pada database

Tabel 4. 2 Definisi Usecase (Lanjutan)

<b>NO</b>	<b>Use Case</b>	<b>Deskripsi</b>
19	Logout	Merupakan suatu proses untuk user agar dapat keluar dari sistem
20	Cetak rekapitulasi	Merupakan suatu proses untuk mencetak hasil pemilihan yang hanya dapat dilakukan oleh Staf kesiswaan
21	Memasukkan kode pemilih	Merupakan suatu proses untuk pemilih agar dapat masuk kedalam sistem dan dapat memilih kandidat yang hanya dapat dilakukan oleh pemilih
22	Memilih kandidat	Merupakan suatu proses untuk memilih kandidat yang hanya dapat dilakukan oleh pemilih
23	Melihat kandidat	Merupakan suatu proses melihat kandidat yang dilakukan oleh pemilih

#### 4.3.3 Skenario *usecase*

Berikut ini adalah skenario jalannya masing-masing *usecase* yang telah didefinisikan sebelumnya :

Nama *usecase* : Menambah pemilih

Skenario : Skenario *usecase* terdapat pada tabel 4.3

Tabel 4. 3 Skenario Usecase Menambah pemilih

<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
<b>Skenario Normal</b>	<ol style="list-style-type: none"> <li>1. memeriksa status login</li> <li>2. memasukkan data pemilih sesuai isian yang ada</li> <li>3. menyimpan data pemilih ke database</li> <li>4. kembali ke <i>form</i> pemilih</li> </ol>

Nama *usecase* : Mengubah pemilih

Skenario : Skenario *usecase* terdapat pada tabel 4.4

Tabel 4. 4 Skenario *usecase* mengubah pemilih

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
1. memeriksa status login	
2. memilih data pemilih yang akan diubah	
	3. menampilkan semua kolom isian berdasarkan data yang sudah dipilih
4. mengubah data pemilih	
	5. menyimpan data pemilih ke database
	6. kembali ke <i>form</i> pemilih

Nama *usecase*: Menghapus pemilih

Skenario: Skenario *usecase* terdapat pada tabel 4.5

Tabel 4. 5 Skenario *usecase* menghapus pemilih

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
1. memeriksa status login	
2. memilih data yang akan dihapus	
	3. menghapus data yang sudah dipilih dari database dan kembali ke <i>form</i> pemilih

Nama *usecase*: Melihat pemilih

Skenario : Skenario *usecase* terdapat pada tabel 4.6

Tabel 4.6 Skenario *usecase* melihat pemilih

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	<ul style="list-style-type: none"> <li>1.memeriksa status login</li> <li>3.menampilkan semua data pemilih yang terdapat di database</li> </ul>

Nama *usecase*: Mencari pemilih

Skenario : Skenario *usecase* terdapat pada tabel 4.7

Tabel 4.7 Skenario *usecase* mencari pemilih

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	<ul style="list-style-type: none"> <li>1.memeriksa status login</li> <li>2. Mengisi nama atau nis pemilih yang ingin dicari</li> <li>3. Mencari data pemilih berdasarkan nis atau nama yang sudah dicari</li> <li>4. Menampilkan data pemilih yang sudah dicari dari database</li> <li>5. Memilih pemilih yang dicari</li> <li>6. Menampilkan detail pemilih yang dipilih</li> </ul>

Aksi Aktor	Reaksi Sistem
<b>Skenario Alternatif</b>	<ul style="list-style-type: none"> <li>1. Mengisi nama atau nis pemilih yang ingin dicari</li> <li>2. Mencari data pemilih berdasarkan nis atau nama yang sudah dicari</li> </ul>

Tabel 4.7 Skenario *usecase* mencari pemilih(Lanjutan)

Aksi Aktor	Reaksi Sistem
	3. Menampilkan data pemilih yang sudah dicari dari database
4. Memilih pemilih yang dicari	5. Menampilkan data pemilih yang dipilih

Nama *usecase*: Mencetak kode pemilih

Skenario: Skenario *usecase* terdapat pada tabel 4.8

Tabel 4.8 Skenario *usecase* mencetak kode pemilih

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	<p>1.memeriksa status login</p> <p>2. Mengisi data pemilih yang ingin dicari</p> <p>3.Mencari data pemilih berdasarkan nis atau nama yang sudah dicari</p> <p>4.memilih data pemilih yang akan dicetak</p> <p>5.menekan <i>button</i> cetak</p> <p>6.mencetak kolom kode login berdasarkan data yang sudah dipilih</p> <p>7. kembali ke <i>form</i> pemilih</p>

Nama *usecase* : Menambah kandidat

Skenario : Skenario *usecase* terdapat pada tabel 4.9

Tabel 4. 9 Skenario Usecase Menambah kandidat

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
1. memeriksa status login 2. Mengisi data kandidat yang ingin dicari dari pemilih 5. Memilih pemilih yang dicari 6. pilih foto kandidat	1.memeriksa status login 2. Mengisi data kandidat yang ingin dicari 3. Mencari data pemilih berdasarkan nis atau nama yang sudah dicari 4. Menampilkan data pemilih yang sudah dicari dari database 7. upload foto kandidat 8. menyimpan data kandidat ke database 9. kembali ke <i>form</i> kandidat

Nama *usecase* : Mengubah kandidat

Skenario : Skenario *usecase* terdapat pada tabel 4.10

Tabel 4. 10 Skenario Usecase Mengubah kandidat

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
1.memeriksa status login 2.memilih data kandidat yang akan diubah 4. mengubah foto kandidat	1.memeriksa status login 2. menampilkan semua kolom isian berdasarkan data yang sudah dipilih 3.menampilkan semua kolom isian berdasarkan data yang sudah dipilih 5. menyimpan data kandidat ke database

Tabel 4. 10 Skenario Usecase Mengubah kandidat (Lanjutan)

Aksi Aktor	Reaksi Sistem
	6. kembali ke <i>form</i> kandidat

Nama *usecase* : Menghapus kandidat

Skenario : Skenario *usecase* terdapat pada tabel 4.11

Tabel 4. 11 Skenario *usecase* menghapus kandidat

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	<ol style="list-style-type: none"> <li>1.memeriksa status login</li> <li>2.memilih data yang akan dihapus</li> <li>3.menghapus data yang sudah dipilih dari database</li> <li>4. kembali ke <i>form</i> kandidat</li> </ol>

Nama *usecase* : Melihat kandidat

Skenario : Skenario *usecase* terdapat pada tabel 4.12

Tabel 4..12 Skenario *usecase* melihat kandidat

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	<ol style="list-style-type: none"> <li>1.memeriksa status login</li> <li>3.menampilkan semua data kandidat yang terdapat di database</li> </ol>

Nama *usecase* : Mencari kandidat

Skenario : Skenario *usecase* terdapat pada tabel 4.13

Tabel 4..13 Skenario *usecase* mencari kandidat

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
2. Mengisi nama atau nis kandidat yang ingin dicari	1. memeriksa status login
5. Memilih kandidat yang dicari	3. Mencari data kandidat berdasarkan nis atau nama yang sudah dicari 4. Menampilkan data kandidat yang sudah dicari dari database 6. Menampilkan detail kandidat yang dipilih
<b>Skenario Alternatif</b>	
1. Mengisi nama atau nis kandidat yang ingin dicari	2. Mencari data kandidat berdasarkan nis atau nama yang sudah dicari 3. Menampilkan data kandidat yang sudah dicari dari database
4. Memilih kandidat yang dicari	5. Menampilkan kandidat yang dipilih

Nama *usecase* : Upload foto kandidat

Skenario : Skenario *usecase* terdapat pada tabel 4.14

Tabel 4..14 Skenario *usecase* upload foto kandidat

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
1.memeriksa status login	
2.memilih foto yang akan dipilih	3.menyimpan foto yang sudah dipilih ke database

Nama *usecase* : Menambah user (panitia MPK)Skenario : Skenario *usecase* terdapat pada tabel 4.15

Tabel 4. 15 Skenario Usecase Menambah user (panitia MPK)

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
1.memeriksa status login	
2.Memilih jenis user mpk	
3. Mengisi nama atau nis user yang ingin dicari	4. Mencari kandidat dari data tabel pemilih berdasarkan nis atau nama yang sudah dicari
6. Memilih pemilih yang dicari	5. Menampilkan data pemilih yang sudah dicari dari database
8.memasukkan username dan password	7. Menampilkan detail pemilih yang dipilih
	9. menyimpan data user ke database
	10. kembali ke <i>form</i> user
<b>Skenario Alternatif</b>	
	1.memeriksa status login

Tabel 4. 15 Skenario Usecase Menambah user (panitia MPK) (Lanjutan)

Aksi Aktor	Reaksi Sistem
2. Memilih jenis user staf 3. memasukkan data staf sesuai isian yang ada	4. menyimpan data user ke database 5. kembali ke <i>form</i> user

Nama *usecase* : Mengubah user (panitia MPK)

Skenario : Skenario *usecase* terdapat pada tabel 4.16

Tabel 4. 16 Skenario Usecase Mengubah user (panitia MPK)

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
2.memilih data user yang akan diubah 4. mengubah data user	1.memeriksa status login 3.menampilkan semua kolom isian bedasarkan data yang sudah dipilih 5. menyimpan data user ke database 6. kembali ke <i>form</i> user

Nama *usecase* : Menghapus user (panitia MPK)

Skenario : Skenario *usecase* terdapat pada tabel 4.17

Tabel 4. 17 Skenario *usecase* menghapus user (panitia MPK)

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
2.memilih data yang akan dihapus	1.memeriksa status login

Tabel 4. 17 Skenario *usecase* menghapus user (panitia MPK) (Lanjutan)

Aksi Aktor	Reaksi Sistem
	3.menghapus data yang sudah dipilih dari database 4. kembali ke <i>form</i> user

Nama *usecase* : Melihat user (panitia MPK)

Skenario : Skenario *usecase* terdapat pada tabel 4.18

Tabel 4..18 Skenario *usecase* melihat user (panitia MPK)

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	1.memeriksa status login 3.menampilkan semua data user yang terdapat di database

Nama *usecase* : Mencari user (panitia MPK)

Skenario : Skenario *usecase* terdapat pada tabel 4.19

Tabel 4.19 Skenario *usecase* mencari user (panitia MPK)

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	1.memeriksa status login 2. Mengisi nama atau ID user yang ingin dicari 3. Mencari data user berdasarkan nis atau nama yang sudah dicari 4. Menampilkan data user yang sudah dicari dari database 5. Memilih user yang dicari

Tabel 4.19 Skenario *usecase* mencari user (panitia MPK) (Lanjutan)

Aksi Aktor	Reaksi Sistem
	6. Menampilkan detail user yang dipilih
<b>Skenario Alternatif</b>	
1. Mengisi nama atau ID user yang ingin dicari	2. Mencari data user berdasarkan nis atau nama yang sudah dicari
	3. Menampilkan data user yang sudah dicari dari database
4. Memilih user yang dicari	5. Menampilkan data user yang dipilih

Nama *usecase* : login user (panitia MPK)

Skenario : Skenario *usecase* terdapat pada tabel 4.20

Tabel 4.20 Skenario *usecase* login user (panitia MPK)

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
1. Memasukkan Username dan Password	2. Memeriksa valid tidaknya data yang dimasukkan dengan memeriksa ke tabel user
	3. Jika valid masuk ke validasi hak akses
	4. masuk halaman administrator
<b>Skenario Alternatif</b>	
1. Memasukkan Username dan Password	

Tabel 4.20 Skenario *usecase* login user (panitia MPK) (Lanjutan)

Aksi Aktor	Reaksi Sistem
<p>4. Memasukkan Username dan Password yang valid</p>	<p>2. Memeriksa valid tidaknya data masukan dengan tabel user</p> <p>3. Jika tidak valid menampilkan pesan Silahkan periksa kembali password dan username anda</p> <p>5. Memeriksa valid tidaknya data masukan</p> <p>6. validasi hak akses</p>

Nama *usecase*: Logout user (panitia MPK)

Skenario: Skenario *usecase* terdapat pada tabel 4.21

Tabel 4.21 Skenario *usecase* logout user (panitia MPK)

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
1.Menekan <i>button</i> Logout	<p>2.melakukan Logout</p> <p>3.masuk ke <i>form</i> utama</p>

Nama *usecase*: Cetak rekapitulasi

Skenario: Skenario *usecase* terdapat pada tabel 4.22

Tabel 4.22 Skenario *usecase* Cetak rekapitulasi

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
	1.Memeriksa status login

Tabel 4.22 Skenario *usecase* Cetak rekapitulasi (Lanjutan)

Aksi Aktor	Reaksi Sistem
2.Menekan <i>button</i> cetak	3.melakukan cetak hasil suara 4.kembali ke <i>form</i> cetak

Nama *usecase*: Memasukkan kode pemilih (Login Pemilih)

Skenario: Skenario *usecase* terdapat pada tabel 4.23

Tabel 4.23 Skenario *usecase* Memasukkan kode pemilih

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
1. Memasukkan Kode pemilih	
	2. Memeriksa valid tidaknya data yang dimasukkan dengan memeriksa ke tabel pemilih
	2. Memeriksa ke variable validasi login sebagai penanda apakah pengguna sudah melakukan login
	3. Mengubah status sudah login
	4. Menampilkan <i>form</i> pemilihan kandidat
<b>Skenario Alternatif</b>	
1. Memasukkan Memasukkan Kode pemilih	
	2. Memeriksa valid tidaknya data masukan dengan tabel user
	3.cek status_login pemilih di database pemilih

Tabel 4.23 Skenario *usecase* Memasukkan kode pemilih (Lanjutan)

Aksi Aktor	Reaksi Sistem
5. Memasukkan Memasukkan Kode pemilih	<p>4. Jika tidak valid menampilkan pesan no login tidak valid atau sudah digunakan</p> <p>6. Memeriksa valid tidaknya data masukan</p> <p>7. masuk halaman pemilihan suara</p>

Nama *usecase*: Memilih kandidat

Skenario: Skenario *usecase* terdapat pada tabel 4.24

Tabel 4.24 Skenario *usecase* Memilih kandidat

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	
3.Memilih kandidat	<p>1. Memeriksa status login</p> <p>2. Menampilkan data kandidat yang ada di database</p> <p>4. Menampilkan detail kandidat yang dipilih</p>
5.Menekan <i>button</i> pilih	<p>6. menyimpan suara ke database kandidat</p> <p>7.melakukan cetak hasil pilihan pemilih</p> <p>8.kembali ke <i>form</i> kode pemilih</p>
<b>Skenario Alternatif</b>	
	1. Memeriksa status login

Tabel 4.24 Skenario *usecase* Memilih kandidat (Lanjutan)

Aksi Aktor	Reaksi Sistem
	2. Menampilkan data kandidat yang ada di database
3. Menekan <i>button</i> tidak memilih	4. melakukan cetak hasil pilihan pemilih 5. kembali ke <i>form</i> kode pemilih

Tabel 4.25 Skenario *usecase* Melihat kandidat

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal</b>	1. memeriksa status login 3. menampilkan semua data user yang terdapat di database

#### 4.4 Pembuatan Aplikasi

Setelah data dan design sudah terkumpul, pada tahap ini dilakukan pengkodean dari sistem informasi e-voting ketua OSIS SMAN 1 PAKUSARI JEMBER. Pada tahap ini perancangan yang dibuat ialah dari sisi administrator dan dari sisi pengguna diantaranya akan dijelaskan pada iterasi 1 proses mengelola pemilih, pada iterasi 2 mencetak kode pemilih, pada iterasi 3 proses mengelola kandidat, pada iterasi 4 proses mengelola user (Panitia MPK), pada iterasi 5 proses login, cek hak akses user, dan logout, pada iterasi 6 proses cetak rekapitulasi suara dari sisi staf, pada iterasi 7 proses login cek status pemilihan dari sisi pemilih, pada iterasi 8. adalah pemilihan kandidat dari sisi pemilih.

##### 4.4.1 Iterasi 1

Pada Iterasi 1, dilakukan pembuatan proses mengelola pemilih pada sisi administrator. Berikut adalah tahapan *prototype* pada pembuatan proses mengelola pemilih.

### a. *Communication*

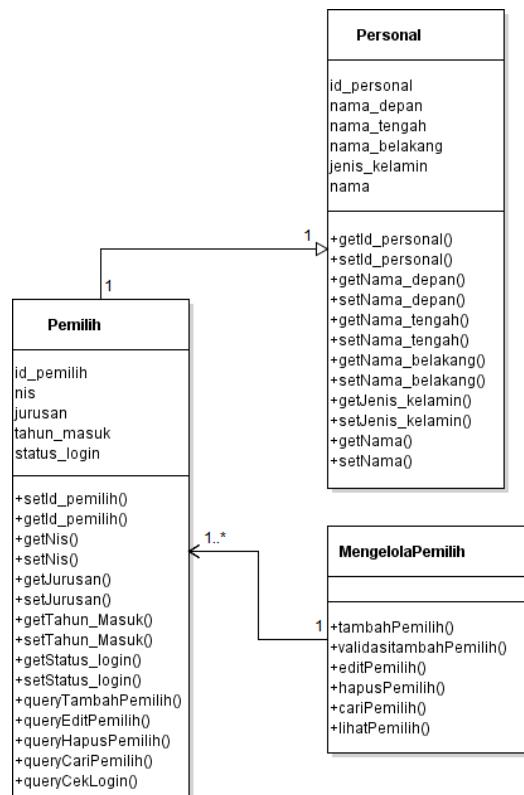
Pada tahap *Communication*, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi administrator. Use Case yang dilakukan adalah menambah pemilih, mengubah pemilih, menghapus pemilih, melihat pemilih, cari pemilih, cetak kode login pemilih.

### b. *Quick Plan*

Pada tahap *Quick Plan*, macam-macam *usecase* yang digunakan adalah:

- 1) Menambah pemilih: Skenario *usecase* mengacu pada tabel 4.3.
- 2) Mengubah pemilih: Skenario *usecase* mengacu pada tabel 4.4.
- 3) Menghapus pemilih: Skenario *usecase* mengacu pada tabel 4.5.
- 4) Melihat pemilih: Skenario *usecase* mengacu pada tabel 4.6.
- 5) Mencari pemilih: Skenario *usecase* mengacu pada tabel 4.7.
- 6) Mencetak kode pemilih: Skenario *usecase* mengacu pada tabel 4.8.

*Class diagram* yang digunakan mengacu pada gambar 4.2

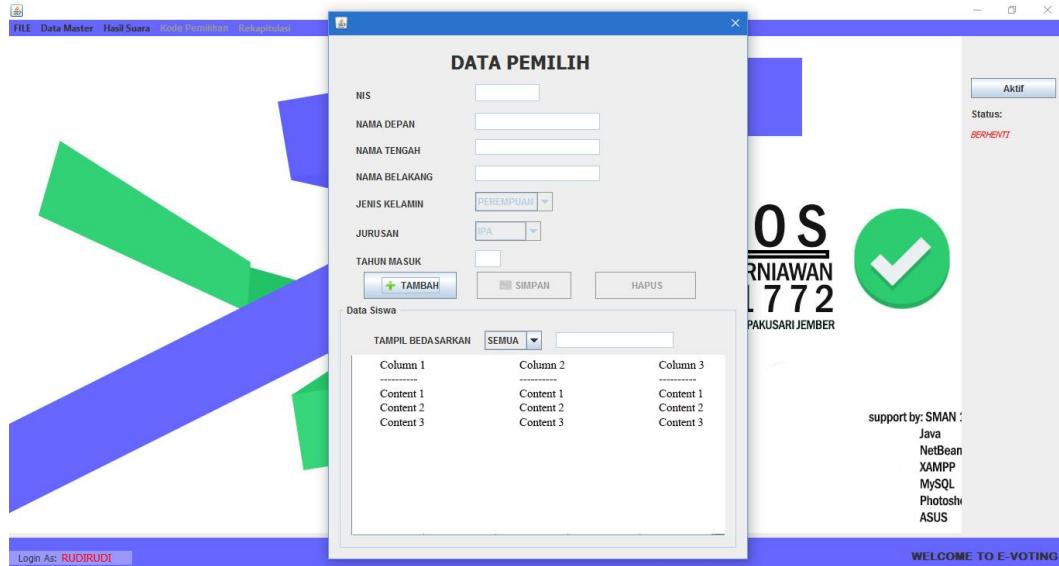


Gambar 4. 2 *Class diagram* mengelola pemilih

### c. Quick Design

Pada tahap ini membangun *mockup* dan design database.

Design *mockup* yang dibangun dapat dilihat pada gambar 4.3



Gambar 4. 3 Design *mockup* mengelola pemilih

Design database yang dibangun adalah tabel pemilih mengacu pada tabel 4.25.

Tabel 4. 26 Tabel pemilih

No	Nama Field	Jenis	Deskripsi
1	ID PEMILIH	varchar(17)	<i>Primary key</i>
2	ID PERSONAL	varchar(17)	<i>Foreign key</i>
3	NIS	varchar(17)	
4	JURUSAN	Smallint	
5	TAHUN_MASUK	varchar(4)	
6	STATUS_LOGIN	smallint	

### d. Construction of Prototype

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun

```
1. public ResultSet show(KEYWORDS keyword, String key) {
2.     try {
3.         String sql = "SELECT * FROM V PEMILIH";
```

```

4.         switch(keyword) {
5.             case NIS:
6.                 sql = sql + " WHERE NIS LIKE '%" + key +"%'";
7.                 break;
8.             case NAMA:
9.                 sql = sql + " WHERE NAMA LIKE '%" +key.trim() + "%'";
10.                }
11.               // System.out.println(sql);
12.               pst = mysql.getConection().prepareStatement(sql);
13.               return(pst.executeQuery());
14.           } catch(java.sql.SQLException e) {
15. System.err.println("Error SQL v_pemilih: " + e.getMessage());
16.         return null;
17.       }
18.     }
19.   public int insert(Pemilih pemilih) {
20.     try {
21.         st = mysql.getConection().createStatement();
22.         String SQL;
23.         SQL = "INSERT INTO PERSONAL VALUES (" + "'" +
24.             pemilih.getId_personal() + "','" +
25.             pemilih.getNama_depan() + "','" +
26.             (pemilih.getNama_tengah() == null?"":") + +
27.             pemilih.getNama_tengah() + "','" + "," +
28.             (pemilih.getNama_belakang()== null?"":") + +
29.             pemilih.getNama_belakang() + "','" + "," +
30.             pemilih.getJenis_kelamin() + ")";
31.         st.addBatch(SQL);
32.         SQL = "INSERT INTO PEMILIH VALUES (" + "'" +
33.             pemilih.getId_pemilih() + "','" +
34.             pemilih.getId_personal() + "','" +
35.             pemilih.getNis() + "','" +
36.             pemilih.getJurusan() + "','" +
37.             pemilih.getTahun_masuk() + "','" +
38.             pemilih.getStatus_login() + "'" +
39.             ")";
30.         st.addBatch(SQL);
41.         int[] count = st.executeBatch();
42.         return count.length;
43.     } catch (SQLException e) {
44. System.err.println("Error Insert Pemilih DI QUERY: " +
45.         e.getMessage());
46.         return 0;
47.     }
48.   }
49.   public int edit(Pemilih pemilih){
50.     try{
51.         st = mysql.getConection().createStatement();
52.         String SQL;
53.         SQL = "UPDATE personal, pemilih SET "+
54. "pemilih.nis=" +""+ pemilih.getNis() +",''+
55. " personal.nama_depan=" +""+pemilih.getNama_depan()
56.             +",''+
57. " personal.nama_tengah=" +"""+pemilih.getNama_tengah()
58.             +",''+
59. " personal.nama_belakang="

```

```

60.          +"'" +pemilih.getNama_belakang() +"' , "+  

61.          " personal.jenis_kelamin=" +"' "+  

62.          pemilih.getJenis_kelamin() +"' , "+  

63.          " pemilih.jurusan=" +"' "+pemilih.getJurusan() +"' , "+  

64.          " pemilih.tahun_masuk=" +"' "+ pemilih.getTahun_masuk()  

65.          +"'" + " WHERE personal.id_personal=pemilih.id_personal  

66.          AND personal.id_personal"+ " = " +pemilih.getId_personal()  

67.          +"'" ;  

68.          st.addBatch(SQL);  

69.          int[] count = st.executeBatch();  

70.          return count.length;           100  

71.      } catch (SQLException e) {  

72.          System.err.println("Error UPDATE Pemilih Di Query!" +  

73.          e.getMessage());  

74.          return 0;  

75.      }  

76.  }  

77.  

78. public int delete(Pemilih pemilih) {  

79.     try {  

80.         st = mysql.getConection().createStatement();  

81.         String SQL;  

82.  

83.         SQL = "DELETE FROM PEMILIH WHERE ID PEMILIH =" +  

84.             pemilih.getId_pemilih();  

85.         st.addBatch(SQL);  

86.         SQL = "DELETE FROM PERSONAL WHERE ID_PERSONAL =" +  

87.             pemilih.getId_personal();  

88.         st.addBatch(SQL);  

89.         int[] count = st.executeBatch();  

90.         return count.length;  

91.  

92.     } catch(java.sql.SQLException e) {  

93.         System.err.println("Error HAPUS PEMILIH SQL:" +  

94.         e.getMessage());  

95.         return 0;  

96.     }  

97. }
98. public boolean ValidasiInsert(String NIS) {
99.     try {
100.         pst = mysql.getConection().prepareStatement("SELECT
101.             COUNT(*) AS NIS FROM PEMILIH WHERE PEMILIH.NIS=?");
102.         pst.setString(1, NIS);
103.         ResultSet rs = pst.executeQuery();
104.         rs.next();
105.         int rowAffected = rs.getInt("NIS");
106.         if (rowAffected == 1) {
107.             System.out.println("Data pemilih Sudah Ada!");
108.             return false;
109.         } else {
110.             return true;
111.         }
112.     } catch (Exception e) {
113.         System.err.println("Error SQL: " + e.getMessage());
114.         return false;
115.     }

```

```

116.      }
117.  public boolean save() {
118.      try {
119.          mysql.getConection().commit();
120.          return true;
121.      } catch (SQLException e) {
122.          try {
123.              mysql.getConection().rollback();
124.          } catch (SQLException e1) {
125.              System.err.println("Error: " + e1.getMessage());
126.          }
127.          System.err.println("Error: " + e.getMessage());
128.          return false;
129.      }
130.  }
131. }
```

#### Kode Program 4.1 model pemilih

Fungsi `ResultSet show (KEYWORDS keyword, String key)` yang digunakan untuk menampilkan seluruh data dari tabel pemilih

Fungsi `insert(Pemilih pemilih)` dengan parameter pemilih digunakan untuk menyimpan data pada variabel `st` yang telah dimasukkan pada FormPemilih ke tabel pemilih.

Fungsi `edit(Pemilih pemilih)` digunakan untuk menyimpan data masukan dari FormPemilih yang dimasukkan ke dalam array kemudian disimpan dalam variabel `st`. `pemilih.getId_personal()` digunakan sebagai id acuan dalam mengubah data pemilih.

Fungsi `delete(Pemilih pemilih)` digunakan untuk menghapus data pada tabel pemilih dengan id yang telah dimasukkan pada variabel `pemilih.getId_pemilih()` dan `pemilih.getId_personal()`.

Fungsi `ValidasiInsert(String NIS)` digunakan untuk mengecek apakah data telah diinputkan ke table pemilih menggunakan parameter NIS

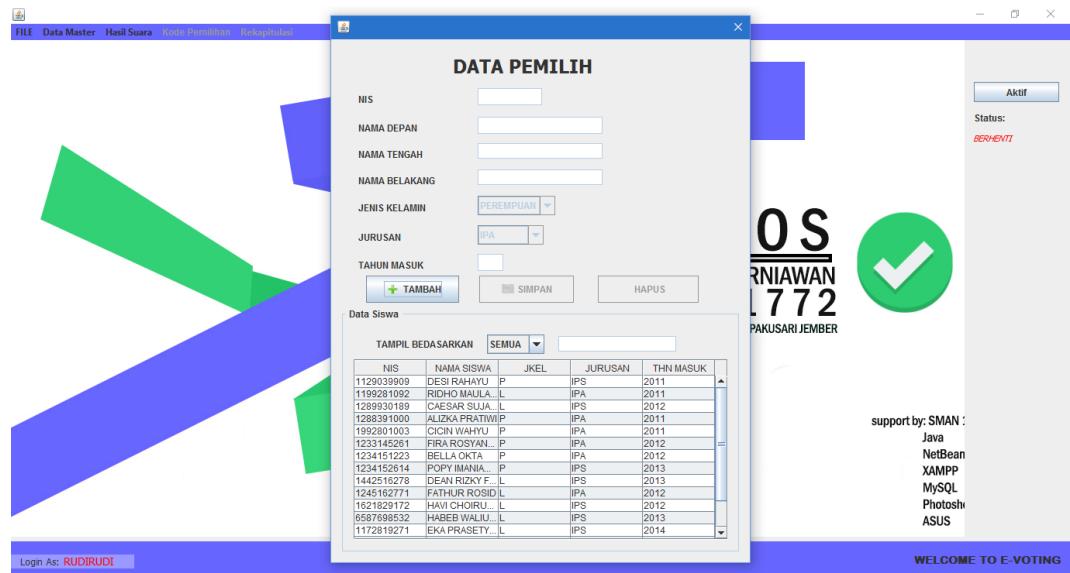
Fungsi `save` digunakan untuk pemanggilan koneksi MySQL yang digunakan untuk menyimpan data masukan dari FormPemilih

#### e. Deployment Delivery and Feedback

Pada tahap ini merupakan tahapan pengujian pada iterasi mengelola pemilih, apabila pengguna tidak sesuai dengan hasil yang di harapkan maka pengembang memperbaiki sistem tersebut sehingga aplikasi atau sistem yang di

buat sesuai dengan yang di inginkan oleh pengguna pengujian ini menggunakan *black box* pada gambar 4.4 adalah Deployment Delivery dan pada tabel 4. 27 adalah pengujian *black box* mengelola pemilih.

### 1) Deployment delivery



Gambar 4.4 Form Pemilih

### 2) Feedback

Tabel 4. 27 Pengujian *black box* mengelola pemilih

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan																																																																					
1	Memasukkan data pemilih dengan benar kemudian klik simpan	<p>DATA PEMILIH</p> <p>NIS: 12345678 NAMA DEPAN: Andi NAMA TENGAH:  NAMA BELAKANG:  JENIS KELAMIN: PEREMPUAN JURUSAN: IPA TAHUN MASUK: 2011</p> <p>TAMPAK BEDASARKAN SEMUA</p> <table border="1"> <thead> <tr> <th>NIS</th> <th>NAMA SISWA</th> <th>JKEL</th> <th>JURUSAN</th> <th>THN MASUK</th> </tr> </thead> <tbody> <tr><td>112903909</td><td>DESI RAHYU</td><td>P</td><td>IPS</td><td>2011</td></tr> <tr><td>1199281092</td><td>RIDHO MAULIA</td><td>L</td><td>IPA</td><td>2011</td></tr> <tr><td>1289930189</td><td>CAESAR SUJA</td><td>L</td><td>IPS</td><td>2012</td></tr> <tr><td>1288391006</td><td>ALIZKA PRATWI</td><td>P</td><td>IPA</td><td>2011</td></tr> <tr><td>1992601003</td><td>CICIN WAHYU</td><td>P</td><td>IPA</td><td>2011</td></tr> <tr><td>1233145261</td><td>FIRAHMAN</td><td>P</td><td>IPA</td><td>2012</td></tr> <tr><td>1234152092</td><td>DEWI AYU KARTIKA</td><td>L</td><td>IPA</td><td>2012</td></tr> <tr><td>1234152614</td><td>POPY IMANAH</td><td>P</td><td>IPS</td><td>2013</td></tr> <tr><td>1442516278</td><td>DEAI RIZKY F</td><td>P</td><td>IPS</td><td>2013</td></tr> <tr><td>1245162771</td><td>FATHUR ROSID</td><td>L</td><td>IPA</td><td>2012</td></tr> <tr><td>1621929172</td><td>HAVI CHORIU</td><td>L</td><td>IPS</td><td>2012</td></tr> <tr><td>6587698532</td><td>HABEB WALIU</td><td>L</td><td>IPS</td><td>2013</td></tr> <tr><td>1172819271</td><td>EKA PRASETY</td><td>L</td><td>IPS</td><td>2014</td></tr> </tbody> </table>	NIS	NAMA SISWA	JKEL	JURUSAN	THN MASUK	112903909	DESI RAHYU	P	IPS	2011	1199281092	RIDHO MAULIA	L	IPA	2011	1289930189	CAESAR SUJA	L	IPS	2012	1288391006	ALIZKA PRATWI	P	IPA	2011	1992601003	CICIN WAHYU	P	IPA	2011	1233145261	FIRAHMAN	P	IPA	2012	1234152092	DEWI AYU KARTIKA	L	IPA	2012	1234152614	POPY IMANAH	P	IPS	2013	1442516278	DEAI RIZKY F	P	IPS	2013	1245162771	FATHUR ROSID	L	IPA	2012	1621929172	HAVI CHORIU	L	IPS	2012	6587698532	HABEB WALIU	L	IPS	2013	1172819271	EKA PRASETY	L	IPS	2014	<p>Sistem sukses menyimpan pemilih dan menampilkan ‘tambah data berhasil’</p>	Valid
NIS	NAMA SISWA	JKEL	JURUSAN	THN MASUK																																																																						
112903909	DESI RAHYU	P	IPS	2011																																																																						
1199281092	RIDHO MAULIA	L	IPA	2011																																																																						
1289930189	CAESAR SUJA	L	IPS	2012																																																																						
1288391006	ALIZKA PRATWI	P	IPA	2011																																																																						
1992601003	CICIN WAHYU	P	IPA	2011																																																																						
1233145261	FIRAHMAN	P	IPA	2012																																																																						
1234152092	DEWI AYU KARTIKA	L	IPA	2012																																																																						
1234152614	POPY IMANAH	P	IPS	2013																																																																						
1442516278	DEAI RIZKY F	P	IPS	2013																																																																						
1245162771	FATHUR ROSID	L	IPA	2012																																																																						
1621929172	HAVI CHORIU	L	IPS	2012																																																																						
6587698532	HABEB WALIU	L	IPS	2013																																																																						
1172819271	EKA PRASETY	L	IPS	2014																																																																						

Tabel 4. 27 Pengujian *black box* mengelola pemilih (Lanjutan)

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
2	Tidak memasukkan data pemilih dengan benar kemudian klik simpan		Sistem sukses menimpan pemilih dan menampilkan data pemilih		Valid
4	Menghapus data pemilih dengan mengeklik data pemilih yang ingin dihapus kemudian pilih button Hapus	Merujuk pada gambar 4.4	Sistem sukses menghapus data pemilih dan menampilkan data pemilih	Merujuk pada gambar 4.4	Valid
5	Mengupdate data pemilih dengan mengeklik data pemilih yang ingin diupdate kemudian klik update	Merujuk pada gambar 4.4	Sistem sukses mengubah pemilih dan menampilkan data pemilih	Merujuk pada gambar 4.4	Valid

#### 4.4.2 Iterasi 2

Pada iterasi 2, dilakukan pembuatan proses mencetak kode pemilihan pemilih pada sisi administrator. Berikut adalah tahapan *prototype* pada pembuatan proses mencetak kode pemilihan pemilih.

##### a. *Communication*

Pada tahap *Communication*, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi administrator. Use Case yang dilakukan adalah mencetak kode pemilihan.

##### b. *Quick Plan*

Pada tahap *Quick Plan*, macam-macam *usecase* yang digunakan adalah:

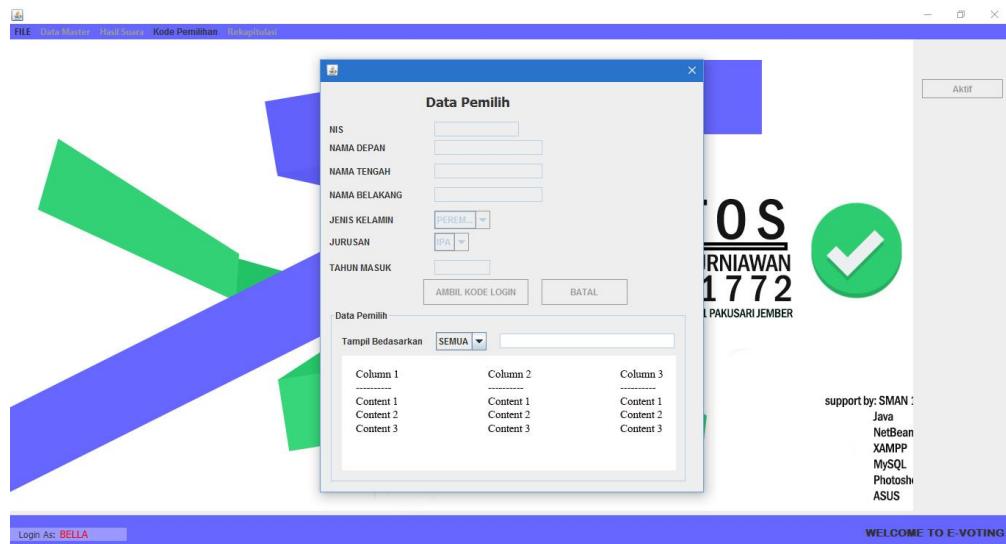
- 1) Mencetak kode pemilih: Skenario *usecase* mengacu pada tabel 4.8.

*Class diagram* yang digunakan mengacu pada gambar 4.2

##### c. *Quick Design*

Pada tahap ini membangun *mockup* dan design database.

Design *mockup* yang dibangun dapat dilihat pada gambar 4.5



Gambar 4. 5 Design *mockup* cetak kode pemilih

Design database yang dibangun adalah tabel pemilih merujuk pada tabel 4.25.

d. *Construction of Prototype*

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun.

```

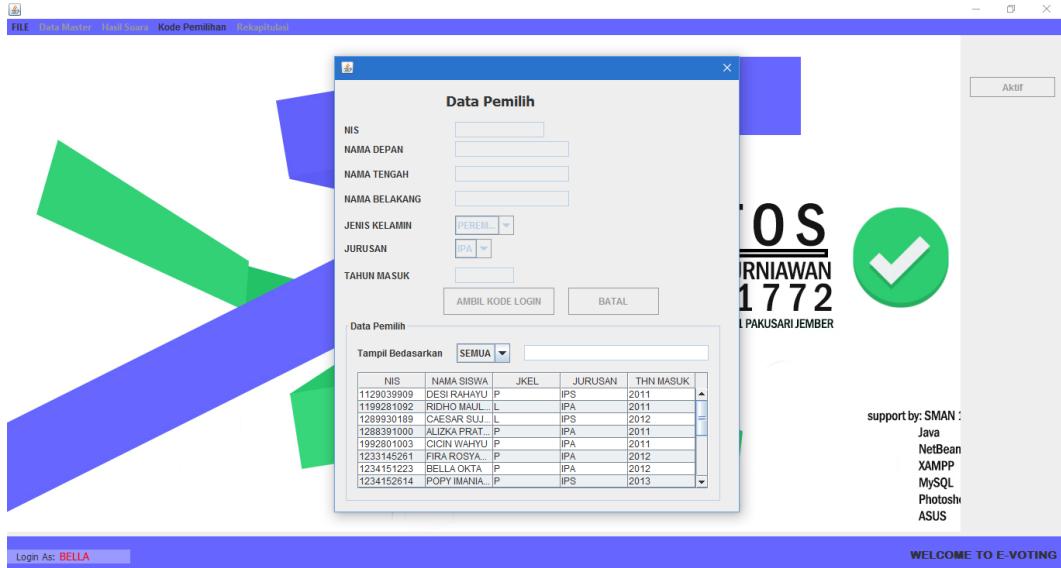
1.     private void btn_printActionPerformed
2.         (java.awt.event.ActionEvent evt) {
3.             // TODO add your handling code here:
4.             try{
5.                 String NamaFile = "./src/evotingserver/sturklogin.jasper";
6.                 Class.forName("com.mysql.jdbc.Driver").newInstance();
7.                 Connection koneksi
8.                 DriverManager.getConnection("jdbc:mysql://localhost/EVOTING"
9.                 ,"root","");
10.                HashMap param = new HashMap();
11.                //Mengambil parameter
12.                param.put("NIS",txt_nis.getText());
13.                JasperPrint JPrint =
14.                JasperFillManager.fillReport(NamaFile,
15.                  param, koneksi);
16.                JasperViewer.viewReport(JPrint);
17.                setComponentsDisplay(COMPONENTS_DISPLAY.UTAMA);
18.            }catch(Exception ex){
19.                System.out.println(ex);
20.            }
21.        }

```

Kode Program 4.2 *form cetak kode pemilih*

Fungsi void btn\_printActionPerformed (java.awt.event. ActionEvent evt) diatas adalah untuk mencetak kode login dengan menggunakan parameter nis untuk mengambil data pemilih.

### 1) Deployment delivery



Gambar 4. 7 form Cetak Kode Pemilih

### 2) Feedback

Tabel 4. 28 Pengujian black box cetak kode pemilih

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan																																																																							
1	Mencetak kode pemilih dengan mengeklik data pemilih yang ingin dicetak kemudian pilih button ambil kode login	<p>Data Pemilih</p> <p>Data Pemilih</p> <p>NIS NAMA DEPAN NAMA TENGAH NAMA BELAKANG JENIS KELAMIN JURUSAN TAHUN MASUK AMBIL KODE LOGIN BATAL</p> <p>Tampil Berdasarkan SEMUA</p> <table border="1"> <thead> <tr> <th>NIS</th> <th>nama siswa</th> <th>JKEL</th> <th>JURUSAN</th> </tr> </thead> <tbody> <tr><td>1129039909</td><td>DESI RAHYAHU</td><td>P</td><td>IPS</td></tr> <tr><td>1199201505</td><td>EDSI RAHAYU</td><td>L</td><td>IPA</td></tr> <tr><td>1238930189</td><td>CAESAR SULI</td><td>L</td><td>IPS</td></tr> <tr><td>1288391000</td><td>ALIZKA PRAT.</td><td>P</td><td>IPA</td></tr> <tr><td>1992801003</td><td>CICIN WAHYU</td><td>P</td><td>IPA</td></tr> <tr><td>1233145261</td><td>FIRA ROSYA</td><td>P</td><td>IPA</td></tr> <tr><td>1234515223</td><td>BELLA OKTA</td><td>P</td><td>IPA</td></tr> <tr><td>1234515214</td><td>POPY MANIA</td><td>P</td><td>IPS</td></tr> </tbody> </table>	NIS	nama siswa	JKEL	JURUSAN	1129039909	DESI RAHYAHU	P	IPS	1199201505	EDSI RAHAYU	L	IPA	1238930189	CAESAR SULI	L	IPS	1288391000	ALIZKA PRAT.	P	IPA	1992801003	CICIN WAHYU	P	IPA	1233145261	FIRA ROSYA	P	IPA	1234515223	BELLA OKTA	P	IPA	1234515214	POPY MANIA	P	IPS	<p>Sistem sukses menyatakan kode sesuai data pemilih</p> <p>Data Pemilih</p> <p>NIS NAMA DEPAN NAMA TENGAH NAMA BELAKANG JENIS KELAMIN JURUSAN TAHUN MASUK AMBIL KODE LOGIN BATAL</p> <p>Tampil Berdasarkan SEMUA</p> <table border="1"> <thead> <tr> <th>NIS</th> <th>nama siswa</th> <th>JKEL</th> <th>JURUSAN</th> </tr> </thead> <tbody> <tr><td>1129039909</td><td>DESI RAHYAHU</td><td>P</td><td>IPS</td></tr> <tr><td>1199201505</td><td>EDSI RAHAYU</td><td>L</td><td>IPA</td></tr> <tr><td>1238930189</td><td>CAESAR SULI</td><td>L</td><td>IPS</td></tr> <tr><td>1288391000</td><td>ALIZKA PRAT.</td><td>P</td><td>IPA</td></tr> <tr><td>1992801003</td><td>CICIN WAHYU</td><td>P</td><td>IPA</td></tr> <tr><td>1233145261</td><td>FIRA ROSYA</td><td>P</td><td>IPA</td></tr> <tr><td>1234515223</td><td>BELLA OKTA</td><td>P</td><td>IPA</td></tr> <tr><td>1234515214</td><td>POPY MANIA</td><td>P</td><td>IPS</td></tr> </tbody> </table>	NIS	nama siswa	JKEL	JURUSAN	1129039909	DESI RAHYAHU	P	IPS	1199201505	EDSI RAHAYU	L	IPA	1238930189	CAESAR SULI	L	IPS	1288391000	ALIZKA PRAT.	P	IPA	1992801003	CICIN WAHYU	P	IPA	1233145261	FIRA ROSYA	P	IPA	1234515223	BELLA OKTA	P	IPA	1234515214	POPY MANIA	P	IPS	Valid
NIS	nama siswa	JKEL	JURUSAN																																																																									
1129039909	DESI RAHYAHU	P	IPS																																																																									
1199201505	EDSI RAHAYU	L	IPA																																																																									
1238930189	CAESAR SULI	L	IPS																																																																									
1288391000	ALIZKA PRAT.	P	IPA																																																																									
1992801003	CICIN WAHYU	P	IPA																																																																									
1233145261	FIRA ROSYA	P	IPA																																																																									
1234515223	BELLA OKTA	P	IPA																																																																									
1234515214	POPY MANIA	P	IPS																																																																									
NIS	nama siswa	JKEL	JURUSAN																																																																									
1129039909	DESI RAHYAHU	P	IPS																																																																									
1199201505	EDSI RAHAYU	L	IPA																																																																									
1238930189	CAESAR SULI	L	IPS																																																																									
1288391000	ALIZKA PRAT.	P	IPA																																																																									
1992801003	CICIN WAHYU	P	IPA																																																																									
1233145261	FIRA ROSYA	P	IPA																																																																									
1234515223	BELLA OKTA	P	IPA																																																																									
1234515214	POPY MANIA	P	IPS																																																																									

#### 4.4.3 Iterasi 3

Pada Iterasi 3, dilakukan pembuatan proses mengelola kandidat pada sisi administrator. Berikut adalah tahapan *prototype* pada pembuatan proses mengelola kandidat.

##### a. *Communication*

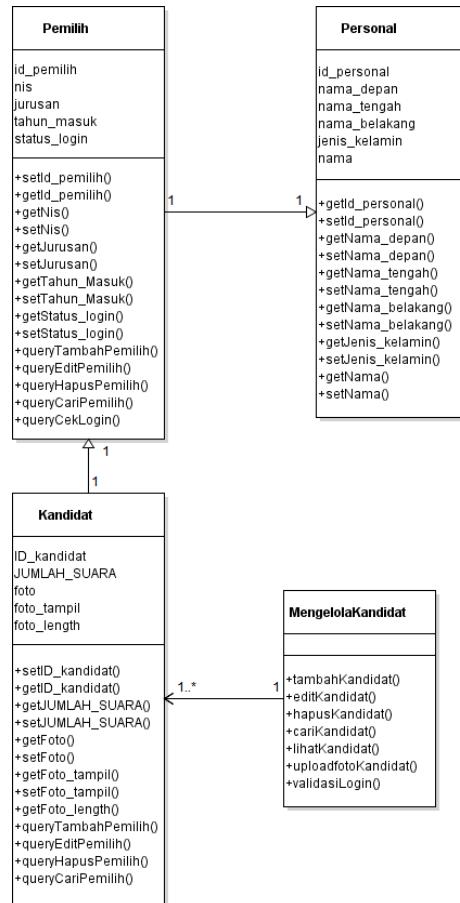
Pada tahap *Communication*, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi administrator. Use Case yang dilakukan adalah menambah kandidat, edit kandidat, menghapus kandidat, melihat kandidat,cari kandidat,upload foto kandidat

##### b. *Quick Plan*

Pada tahap *Quick Plan*,macam-macam *usecase* yang digunakan adalah :

- 1) Menambah kandidat : Skenario *usecase* yang digunakan mengacu pada tabel 4.9.
- 2) Mengubah kandidat : Skenario *usecase* yang digunakan mengacu pada tabel 4.10.
- 3) Menghapus kandidat : Skenario *usecase* yang digunakan mengacu pada tabel 4.11.
- 4) Melihat kandidat : Skenario *usecase* yang digunakan mengacu pada tabel 4.12.
- 5) Mencari kandidat : Skenario *usecase* yang digunakan mengacu pada tabel 4.13.
- 6) Upload foto kandidat : Skenario *usecase* yang digunakan mengacu pada tabel 4.14.

*Class diagram* yang digunakan mengacu pada gambar 4.7



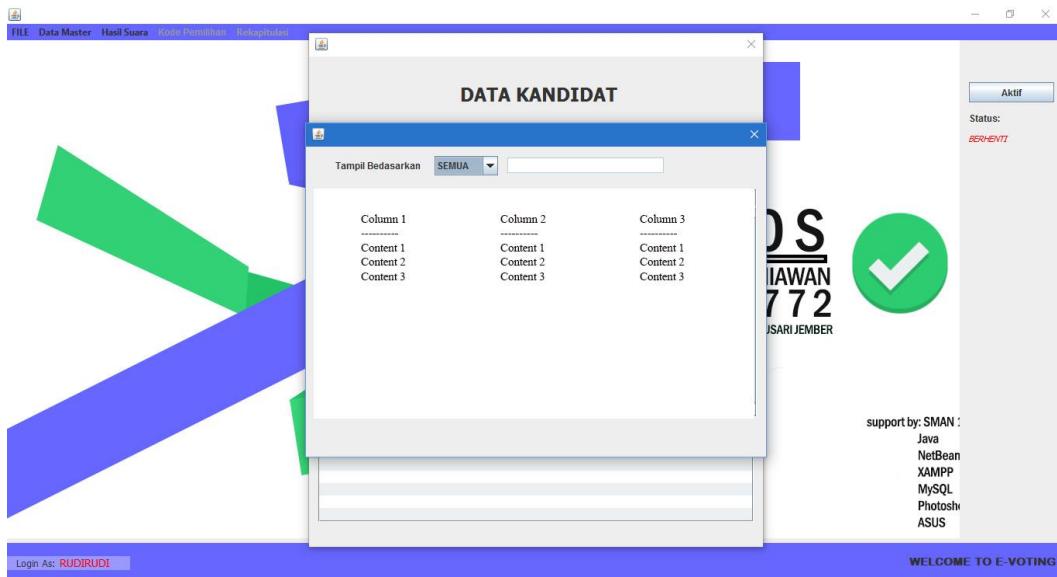
Gambar 4. 7 *Class diagram* mengelola kandidat

### c. Quick Design

Pada tahap ini membangun *mockup* dan design database. Design *mockup* yang dibangun dapat dilihat pada gambar 4.8 dan gambar 4.9



Gambar 4. 8 Design *mockup* mengelola kandidat



Gambar 4. 9 Design *mockup* mencari calon kandidat dari data pemilih

Design database yang dibangun menggunakan tabel kandidat yang merupakan table turunan dari tabel pemilih dan personal

Tabel 4. 29 Tabel kandidat

No	Nama Field	Jenis	Deskripsi
1	ID_KANDIDAT	varchar(17)	<i>Primary key</i>
2	ID PEMILIH	varchar(17)	<i>Foreign key</i>
3	FOTO	Blob	
4	JUMLAH_SUARA	int(3)	

#### d. *Construction of Prototype*

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun.

```

1.    public ResultSet show(KEYWORDS keyword, String key) {
2.        try {
3.            String sql = "SELECT * FROM V_KANDIDAT";
4.            pst = mysql.getConection().prepareStatement(sql);
5.            return(pst.executeQuery());
6.        } catch(java.sql.SQLException e) {
7.            System.err.println("Error QUERY v_kandidat: "
8.                + e.getMessage());
9.            return null;
10.       }

```

```

11.    }
12.    public int insert(Kandidat kandidat) {
13.        try {
14.            pst =mysql.getConection().prepareStatement
15.                ("INSERT INTO KANDIDAT VALUES (?,?,?,?,?)");
16.                pst.setString(1,kandidat.getID_kandidat());
17.                pst.setString(2,kandidat.getId_pemilih());
18.                pst.setBinaryStream(3, kandidat.getFoto()
19.                    ,kandidat.getFoto_Length());
20.                pst.setInt(4, kandidat.getJUMLAH_SUARA());
21.                int count = pst.executeUpdate();
22.                return count;
23.        } catch (SQLException e) {
24.            System.err.println("Error Insert Pemilih Di SQUERY: "
25.                + e.getMessage());
26.            return 0;
27.        }
28.    }
29.    public int edit(Kandidat kandidat){
30.        try{
31.            pst =mysql.getConection().prepareStatement
32.                ("UPDATE KANDIDAT SET FOTO=? WHERE ID_KANDIDAT=?");
33.                pst.setBinaryStream(1, kandidat.getFoto()
34.                    ,kandidat.getFoto_Length());
35.                pst.setString(2,kandidat.getID_kandidat());
36.                int count = pst.executeUpdate();
37.                return count;
38.        } catch (SQLException e) {
39.            System.err.println("Error UPDATE FOTO Kandidat Di Query! "
40.                + e.getMessage());
41.            return 0;
42.        }
43.    }
44.    public int delete(Kandidat kan) {
45.        try {
46.            st = mysql.getConection().createStatement();
47.            String SQL;
48.            SQL = "DELETE FROM KANDIDAT WHERE ID_KANDIDAT = "
49.                + kan.getID_kandidat();
50.            st.addBatch(SQL);
51.            int[] count = st.executeBatch();
52.            return count.length;
53.        }catch(java.sql.SQLException e) {
54.            System.err.println("Error HAPUS KANDIDAT DI QUERY: "
55.                + e.getMessage());
56.            return 0;
57.        }
58.    }
59.    public boolean ValidasiInsert(String ID PEMILIH) {
60.        try {
61.            pst = mysql.getConection().prepareStatement
62.                ("SELECT COUNT(*) AS ID FROM KANDIDAT
63.                 WHERE KANDIDAT.ID PEMILIH=?");
64.                pst.setString(1, ID PEMILIH);
65.                ResultSet rs = pst.executeQuery();
66.                rs.next();

```

```

67.         int rowAffected = rs.getInt("ID");
68.         if (rowAffected == 1) {
69.             System.out.println("Data Kandidat Sudah Ada!");
70.             return false;
71.         }else {
72.             return true;
73.         }
74.     } catch (Exception e) {
75.         System.err.println("Error SQL: " + e.getMessage());
76.         return false;
77.     }
78. }
79. public boolean save() {
80.     try {
81.         mysql.getConection().commit();
82.         return true;
83.     }catch (SQLException e) {
84.         try {
85.             mysql.getConection().rollback();
86.         }catch (SQLException e1) {
87.             System.err.println("Error: " + e1.getMessage());
88.         }
89.         System.err.println("Error: " + e.getMessage());
90.         return false;
91.     }

```

#### Kode Program 4.3 model kandidat

Fungsi `ResultSet show (KEYWORDS keyword, String key)` yang digunakan untuk menampilkan seluruh data dari tabel kandidat

Fungsi `insert(Kandidat kandidat)` dengan parameter kandidat digunakan untuk menyimpan data pada variabel `st` yang telah dimasukkan pada `FormKandidat` ke tabel kandidat.

Fungsi `edit(Kandidat kandidat)` digunakan untuk menyimpan data masukan dari `FormKandidat` yang dimasukkan ke dalam array kemudian disimpan dalam variabel `st`. data yang bisa diubah di kandidat adalah foto, `kandidat.getID_kandidat()` digunakan sebagai id acuan dalam mengubah data kandidat.

Fungsi `delete(Kandidat kan)` digunakan untuk menghapus data pada tabel kandidat dengan `id` yang telah dimasukkan pada variabel `kant.getId_kandidat()`.

Fungsi `ValidasiInsert(String ID PEMILIH)` digunakan untuk mengecek apakah data telah diinputkan ke table kandidat menggunakan parameter `ID PEMILIH`

Fungsi `save` digunakan untuk pemanggilan koneksi MySQL yang digunakan untuk menyimpan data masukan dari `FormKandidat`

```

1. private void btnbrowseActionPerformed
2.   (java.awt.event.ActionEvent evt) {
3.     // TODO add your handling code here:
4.     JFileChooser chooser = new JFileChooser
5.     (System.getProperty("Documents/image TA"));
6.     chooser.setFileSelectionMode(JFileChooser.FILES_ONLY);
7.     chooser.setFileFilter(new FileNameExtensionFilter
8.       ("jpg|png|bmp","jpg","png","bmp"));
9.     if(chooser.showOpenDialog(this)==
10.       JFileChooser.APPROVE_OPTION){
11.         file=chooser.getSelectedFile();
12.         try {
13.           image = ImageIO.read(file);
14.           Image myImg =image.getScaledInstance(lbl_gmbr.getWidth(),
15.             lbl_gmbr.getHeight(),Image.SCALE_SMOOTH);
16.           ImageIcon imageIcon=new ImageIcon(myImg);
17.           lbl_gmbr.setIcon(imageIcon);
18.         } catch (IOException ex) {
19.           Logger.getLogger(FormKandidat.class.getName())
20.             .log(Level.SEVERE, null, ex);
21.         }
22.         txtfoto.setText(file.getAbsolutePath());
23.       }
24.         btnsimpan.setEnabled(true);
25.         btnhapus.setEnabled(false);
26.     }

```

#### Kode Program 4.4 kode upload foto

Fungsi btnbrowseActionPerformed digunakan untuk memanggil jfilechoser yang digunakan untuk memilih foto yang kemudian foto diarahkan ke lbl\_gambar untuk ditampilkan.

```

1. private void btncari_kandidatActionPerformed
2. (java.awt.event.ActionEvent evt) {
3.   // TODO add your handling code here:
4.   if (caridata == true) {
5.     JFrame frame = (JFrame)
6.       SwingUtilities.getWindowAncestor(this);
7.     FormCariKandidat formcari_Pemilih
8.       = new FormCariKandidat(frame,true);
9.     formcari_Pemilih.setLocationRelativeTo(null);
10.    formcari_Pemilih.setFormKandidat(this);
11.    formcari_Pemilih.setModecari();
12.    formcari_Pemilih.setParent(this);
13.    formcari_Pemilih.setVisible(true);
14.    if(txtnama.getText().length()==0){
15.      JOptionPane.showMessageDialog(rootPane,
16.        "HARAP PILIH KANDIDAT");
17.    }else{
18.      setComponentsDisplay
19.        (COMPONENTS_DISPLAY.SEBELUM_TAMBAH_ATAU_BATAL);
20.      caridata = false;
21.    }

```

#### Kode Program 4.5 kode mencari kandidat dari data pemilih

Fungsi btncari\_kandidatActionPerformed digunakan untuk memanggil form cari kandidat dari data pemilih.

```

29     -----
30     public void mousePressed(java.awt.event.MouseEvent evt) {
31         if (zTable1.isRecordSelected() && evt.getClickCount()==2 && isSearchActive) {
32             String Id_pemilih = tm_pemilih.getValueParam(zTable1.getSelectedRow(), TableModel_Pemilih.ID PEMILIH).toString();
33             String NIS = tm_pemilih.getValueParam(zTable1.getSelectedRow(), TableModel_Pemilih.NIS).toString();
34             String NAMA = tm_pemilih.getValueParam(zTable1.getSelectedRow(), TableModel_Pemilih.NAMA).toString();
35             String NAMA_DEPAN = tm_pemilih.getValueParam(zTable1.getSelectedRow(), TableModel_Pemilih.NAMA_DEPAN).toString();
36             String NAMA_TENGAH = tm_pemilih.getValueParam(zTable1.getSelectedRow(), TableModel_Pemilih.NAMA_TENGAH).toString();
37             String NAMA_BELAKANG = tm_pemilih.getValueParam(zTable1.getSelectedRow(), TableModel_Pemilih.NAMA_BELAKANG).toString();
38             String JURUSAN_KANDIDAT = tm_pemilih.getValueParam(zTable1.getSelectedRow(), TableModel_Pemilih.JURUSAN).toString();
39             String JENIS_KELAMIN = tm_pemilih.getValueParam(zTable1.getSelectedRow(), TableModel_Pemilih.JENIS_KELAMIN).toString();
40             formkandidat.setId_pemilih(Id_pemilih);
41             formkandidat.setNIS_KANDIDAT(NIS);
42             formkandidat.setNAMA(NAMA);
43             formkandidat.setNAMA_DEPAN(NAMA_DEPAN);
44             formkandidat.setNAMA_TENGAH(NAMA_TENGAH);
45             formkandidat.setNAMA_BELAKANG(NAMA_BELAKANG);
46             formkandidat.setJURUSAN_KANDIDAT(Integer.parseInt(JURUSAN_KANDIDAT));
47         }
48     }

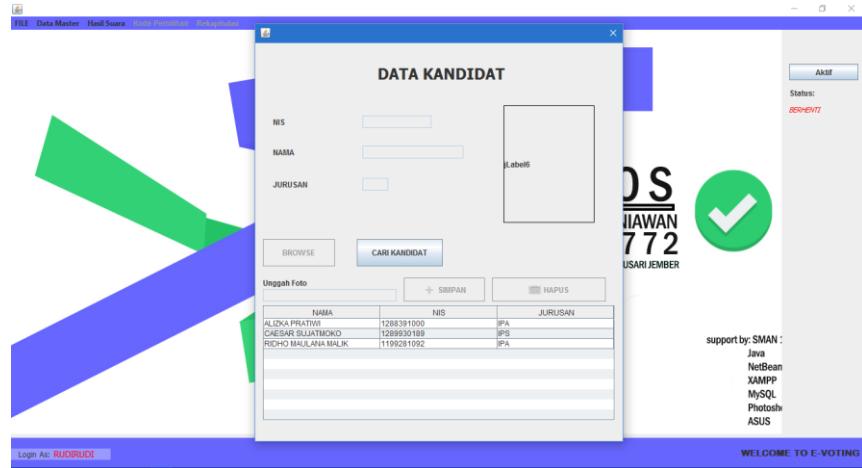
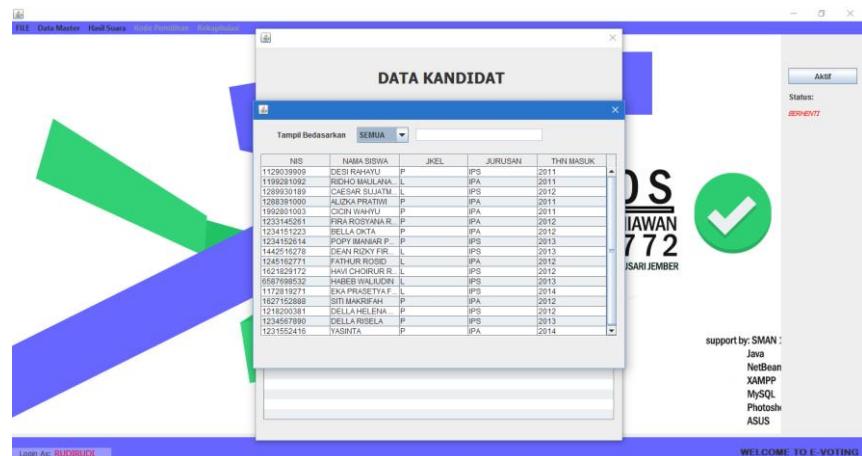
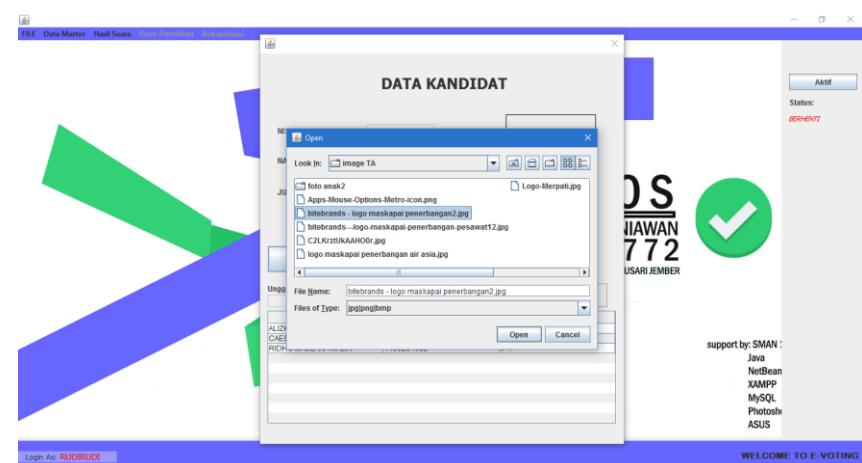
```

Gambar 4. 10 Script mengambil data kandidat dari data pemilih

Fungsi mousePressed digunakan untuk mengambil data pemilih dengan cara double clik pada mose yang kemudian disimpan di variable, variable lalu dikirimkan ke form kandidat

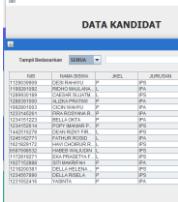
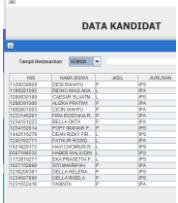
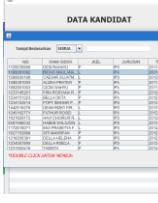
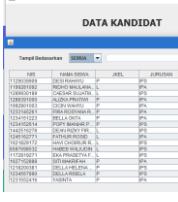
#### e. Deployment Delivery and Feedback

Pada tahap ini merupakan tahapan pengujian pada iterasi mengelola kandidat, apabila pengguna tidak sesuai dengan hasil yang di harapkan maka pengembang memperbaiki sistem tersebut sehingga aplikasi atau sistem yang di buat sesuai dengan yang di inginkan oleh pengguna pengujian ini menggunakan *black box* pada gambar 4.11, gambar 4.12, gambar 4.13 adalah Deployment Delivery dan pada tabel 4.30 adalah pengujian *black box* mengelola kandidat

Gambar 4. 11 *form kandidat*Gambar 4. 12 *form cari kandidat dari data pemilih*Gambar 4. 13 *form cari foto*

## 2) Feedback

Tabel 4. 30 Pengujian black box mengelola kandidat

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Menutup form cari kandidat atau tidak memilih dari data pemilih		Sistem sukses memunculkan peringatan "harap pilih kandidat"		Valid
2	Tidak menggunakan double click pada saat memilih kandidat		Sistem sukses memunculkan peringatan "Double klik untuk memilih"		Valid
3	Mencari kandidat dengan benar dan menggunakan double click pada saat memilih kandidat		Sistem sukses mengambil data sesuai dari data pemilih		Valid

Tabel 4. 30 Pengujian *black box mengelola kandidat* (Lanjutan)

No	Skenario	Test Case	Hasil yang diharapkan	Hasil	Kesimpulan									
4	Memasukkan data kandidat dengan benar kemudian klik simpan		Sistem sukses menyimpan kandidat sesuai data dari pemilih	 <table border="1"> <tr> <td>1234567890</td> <td>NIS</td> <td>IPK</td> </tr> <tr> <td>1234567890</td> <td>Nama</td> <td>IPK</td> </tr> <tr> <td>1234567890</td> <td>1234567890</td> <td>IPK</td> </tr> </table>	1234567890	NIS	IPK	1234567890	Nama	IPK	1234567890	1234567890	IPK	Valid
1234567890	NIS	IPK												
1234567890	Nama	IPK												
1234567890	1234567890	IPK												
5	Menghapus data kandidat dengan mengeklik data kandidat yang ingin dihapus kemudian pilih button Hapus	Merujuk pada gambar 4.15	Sistem sukses menghapus data pemilih dan menampilkan data pemilih	Merujuk pada gambar 4.15	Valid									
8	Mengupdate data kandidat dengan mengeklik data kandidat yang ingin diupdate kemudian klik update	Merujuk pada gambar 4.15	Sistem sukses mengubah foto kandidat dan menampilkan data kandidat	Merujuk pada gambar 4.15	Valid									

#### 4.4.4 Iterasi 4

Pada Iterasi 4, dilakukan pembuatan proses mengelola user pada sisi administrator. Berikut adalah tahapan *prototype* pada pembuatan proses mengelola user.

##### a. Communication

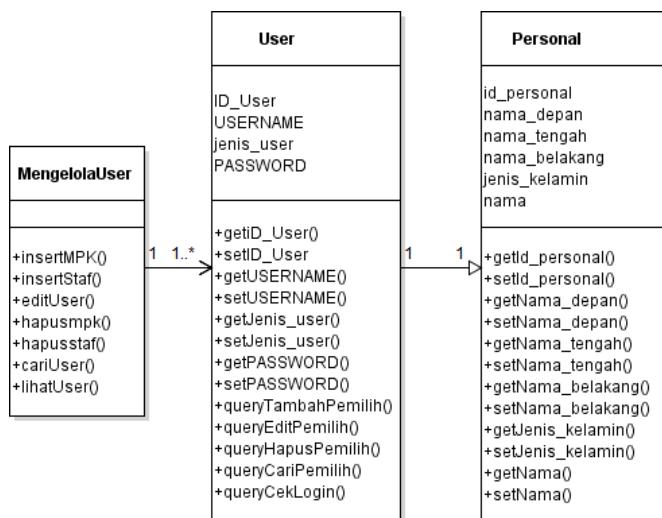
Pada tahap Communication, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi administrator. Use Case yang dilakukan adalah menambah user, edit user, menghapus user, melihat user,cari user.

##### b. Quick Plan

Pada tahap *Quick Plan*, macam-macam *usecase* yang digunakan adalah :

- 1) Menambah user : Skenario *usecase* yang digunakan mengacu pada tabel 4.15.
- 2) Mengubah user : Skenario *usecase* yang digunakan mengacu pada tabel 4.16.
- 3) Menghapus user : Skenario *usecase* yang digunakan mengacu pada tabel 4.17.
- 4) Melihat user: Skenario *usecase* yang digunakan mengacu pada tabel 4.18.
- 5) Mencari user: Skenario *usecase* yang digunakan mengacu pada tabel 4.19.

*Class diagram* yang digunakan mengacu pada gambar 4.14

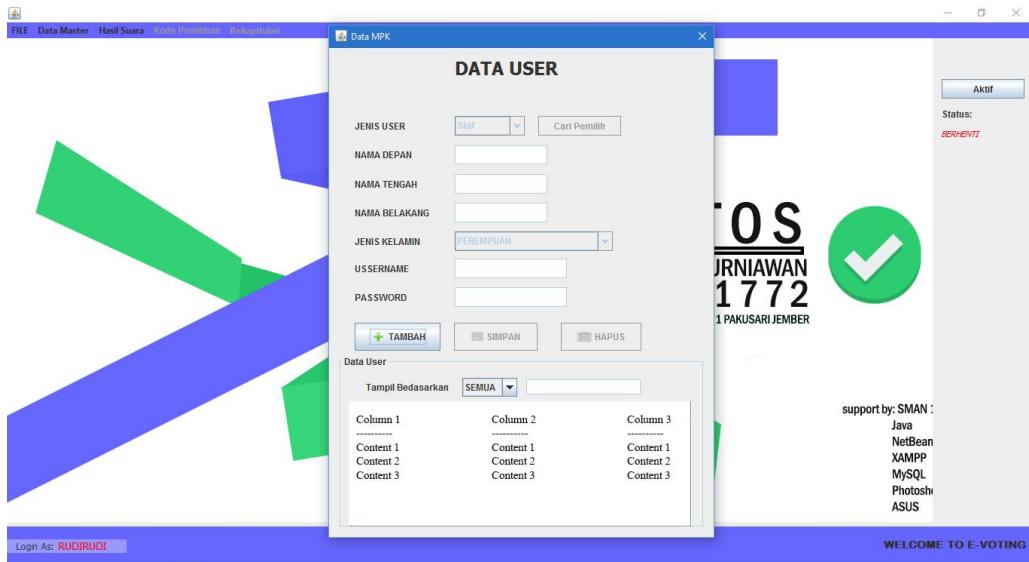


Gambar 4. 14 *Class diagram* mengelola user

##### c. Quick Design

Pada tahap ini membangun *mockup* dan design database.

Design *mockup* yang dibangun dapat dilihat pada gambar 4.15



Gambar 4. 15 *Design mockup* mengelola user

Design database yang dibangun adalah tabel USER mengacu pada tabel 4.29.

Tabel 4. 31 Tabel USER

No	Nama Field	Jenis	Deskripsi
1	ID_USER	varchar(17)	<i>Primary key</i>
2	ID_PERSONAL	varchar(17)	<i>Foreign key</i>
3	USERNAME	varchar(20)	
4	PASSWORD	varchar(20)	
5	JENIS_USER	smallint(6)	

#### d. *Construction of Prototype*

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun.

```

1.     public ResultSet show(CUser.KEYWORDS keyword, String key) {
2.         try {
3.             String sql = "SELECT * FROM V_USER";
4.             switch(keyword) {
5.                 case ID_USER:
6.                     sql = sql + " WHERE ID_USER LIKE '%" + key + "%'";
7.                     break;

```

```

8.         case NAMA:
9.             sql = sql + " WHERE NAMA LIKE '%" +
10.                 key.trim() + "%'";
11.             }
12.             pst = mysql.getConection().prepareStatement(sql);
13.             return(pst.executeQuery());
14.         } catch(java.sql.SQLException e) {
15.             System.err.println("Error SQL v_mpk: " + e.getMessage());
16.             return null;
17.         }
18.     }
19.     public int insertStaf(User user) {
20.         try {
21.             st = mysql.getConection().createStatement();
22.             String SQL;
23.             SQL = "INSERT INTO PERSONAL VALUES (" + "'" +
24.                 user.getId_personal() + "','" +
25.                 user.getNama_depan() + "','" +
26.                 (user.getNama_tengah() == null?"":") +
27.                     + user.getNama_tengah() + "") + "," +
28.                     (user.getNama_belakang()== null?"":") +
29.                     user.getNama_belakang() + ")" + "," +
30.                     user.getJenis_kelamin() +
31.                     ")";
32.             st.addBatch(SQL);
33.             SQL = "INSERT INTO USER VALUES (" + "'" +
34.                 user.getID_User() + "','" +
35.                 user.getId_personal() + "','" +
36.                 user.getUsername() + "','" +
37.                 user.getPassword() + "','" +
38.                 user.getJenis_user() + ")" +
39.                 ")";
40.             st.addBatch(SQL);
41.             int[] count = st.executeBatch();
42.             return count.length;
43.         } catch (SQLException e) {
44.             System.err.println("Error Insert data staf DI QUERY: " +
45.                 + e.getMessage());
46.             return 0;
47.         }
48.     }
49.     public int insertMpk(User user) {
50.         try {
51.             st = mysql.getConection().createStatement();
52.             String SQL;
53.             SQL = "INSERT INTO USER VALUES (" + "'" +
54.                 user.getID_User() + "','" +
55.                 user.getId_personal() + "','" +
56.                 user.getUsername() + "','" +
57.                 user.getPassword() + "','" +
58.                 user.getJenis_user() + ")" +
59.                 ";
60.             st.addBatch(SQL);
61.             int[] count = st.executeBatch();
62.             return count.length;
63.         } catch (SQLException e) {

```

```

64.         System.err.println("Error Insert MPK DI QUERY: " +
65.             + e.getMessage());
66.         return 0;
67.     }
68. }
69. public int edit(User user) {
70.     try{
71.         st = mysql.getConection().createStatement();
72.         String SQL;
73.         SQL = "UPDATE personal, user SET " +
74.             "USER.id_user=" +'"'+ user.getID_User() +"' , " +
75.             " personal.nama_depan=" +'"'+
76.                 user.getNama_depan() +"' , " +
77.             " personal.nama_tengah=" +
78.                 +'"'+user.getNama_tengah() +"' , " +
79.             " personal.nama_belakang=" +
80.                 +'"'+user.getNama_belakang() +"' , " +
81.             " personal.jenis_kelamin=" +
82.                 +'"'+user.getJenis_kelamin() +"' , " +
83.             " user.jenis_user=" +
84.                 +'"'+user.getJenis_user() +"' , " +
85.             " user.username=" +'"'+user.getUsername() +"' , " +
86.             " user.password=" +'"'+user.getPassword() +"' +
87.             " WHERE personal.id_personal=user.id_personal
88.                 AND personal.id_personal" + "=" +
89.                     user.getId_personal() +";";
90.         st.addBatch(SQL);
91.         int[] count = st.executeBatch();
92.         return count.length;
93.     } catch (SQLException e) {
94.         System.err.println("Error UPDATE Pemilih Di Query!" +
95.             + e.getMessage());
96.         return 0;
97.     }
98. }
99.
100.    public int deleteStaf(User user) {
101.        try {
102.            st = mysql.getConection().createStatement();
103.            String SQL;
104.
105.            SQL = "DELETE FROM USER WHERE ID_USER = " +
106.                user.getID_User();
107.            st.addBatch(SQL);
108.            SQL = "DELETE FROM PERSONAL WHERE ID_PERSONAL = " +
109.                user.getId_personal();
110.            st.addBatch(SQL);
111.            int[] count = st.executeBatch();
112.            return count.length;
113.
114.        } catch(java.sql.SQLException e) {
115.            System.err.println("Error HAPUS USER DI SQL: " +
116.                e.getMessage());
117.            return 0;
118.        }
119.    }

```

```

110.     public int deleteMPK(User user) {
111.         try {
112.             st = mysql.getConection().createStatement();
113.             String SQL;
114.             SQL = "DELETE FROM USER WHERE ID_USER = " +
115.                 user.getID_User();
116.             st.addBatch(SQL);
117.             int[] count = st.executeBatch();
118.             return count.length;
119.         } catch(java.sql.SQLException e) {
120.             System.err.println("Error HAPUS USER DI SQL: " +
121.                 e.getMessage());
122.             return 0;
123.         }
124.     }
125.
126.     public boolean ValidasiInsert(String ID_PERSONAL) {
127.         try {
128.             pst = mysql.getConection().prepareStatement ("SELECT
129.                 COUNT(*) AS ID FROM USER WHERE USER.ID_PERSONAL=?");
130.             pst.setString(1, ID_PERSONAL);
131.             ResultSet rs = pst.executeQuery();
132.             rs.next();
133.             int rowAffected = rs.getInt("ID");
134.             if (rowAffected == 1) {
135.                 System.out.println("Data User Sudah Ada!");
136.                 return false;
137.             } else {
138.                 return true;
139.             }
140.
141.         } catch (Exception e) {
142.             System.err.println("Error SQL: " + e.getMessage());
143.             return false;
144.         }
145.     }
146.     public boolean save() {
147.         try {
148.             mysql.getConection().commit();
149.             return true;
150.         } catch (SQLException e) {
151.             try {
152.                 mysql.getConection().rollback();
153.             } catch (SQLException e1) {
154.                 System.err.println("Error: " +
155.                     e1.getMessage());
156.             }
157.             System.err.println("Error: " + e.getMessage());
158.             return false;
159.         }
160.     }

```

#### Kode Program 4.6 model user

Fungsi `ResultSet show (KEYWORDS keyword, String key)` yang digunakan untuk menampilkan seluruh data dari tabel user

Fungsi insertStaf(User user) dengan parameter user digunakan untuk menyimpan data pada variabel st yang telah dimasukkan pada FormUser ke tabel personal dan user.

Fungsi insertMPK(User user) dengan parameter user digunakan untuk menyimpan data pada variabel st yang telah dimasukkan pada FormUser hanya ke table user karena insert MPK mengambil dari data pemilih.

Fungsi edit(User user) digunakan untuk menyimpan data masukan dari FormUser yang dimasukkan ke dalam array kemudian disimpan dalam variabel st., data yang bisa di edit jika MPK adalah username dan password jika jenis User Staf bisa mengubah semua data ,personal.getID\_ personal digunakan sebagai id acuan dalam mengubah data kandidat.

Fungsi deleteStaf(User user) digunakan untuk menghapus data pada tabel user dan personal dengan id yang telah dimasukkan pada variabel user.getId\_personal dan user.getID\_User().

Fungsi deleteMPK(User user) digunakan untuk menghapus data pada tabel user saja dengan id yang telah dimasukkan pada variabel user.getID\_User().

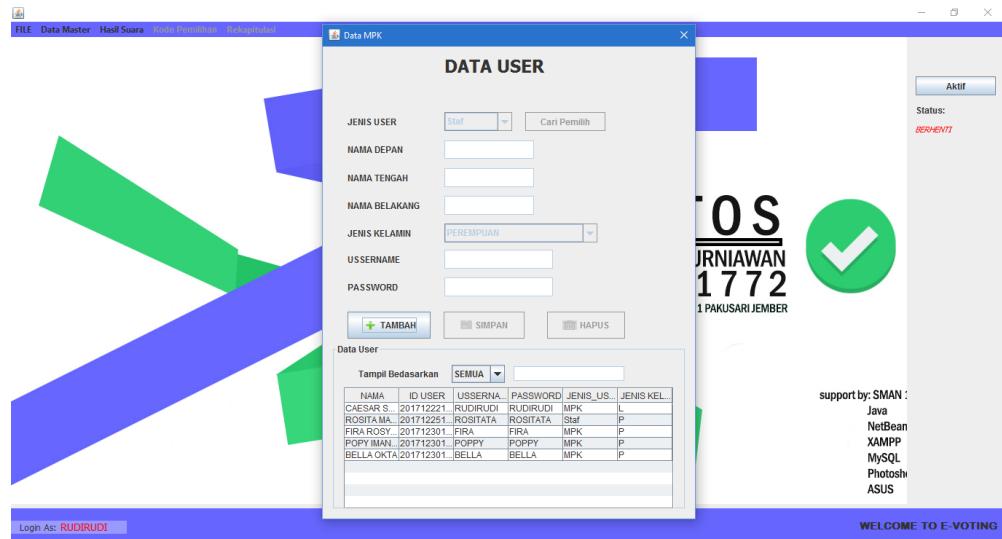
Fungsi ValidasiInsert(String ID\_PERSONAL) digunakan kusus jenis user Staf untuk mengecek apakah data telah diinputkan ke table user menggunakan parameter ID\_PERSONAL ke tabel personal

Fungsi save digunakan untuk pemanggilan koneksi MySQL yang digunakan untuk menyimpan data masukan dari FormUser.

#### e. Deployment Delivery and Feedback

Pada tahap ini merupakan tahapan pengujian pada iterasi mengelola user, apabila pengguna tidak sesuai dengan hasil yang di harapkan maka pengembang memperbaiki sistem tersebut sehingga aplikasi atau sistem yang di buat sesuai dengan yang di inginkan oleh pengguna pengujian ini menggunakan *black box* pada gambar 4.16 adalah Deployment Delivery dan pada tabel 4.32 adalah pengujian *black box* mengelola user.

### 1) Deployment delivery



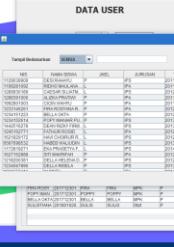
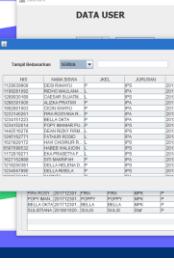
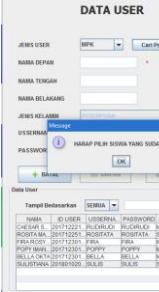
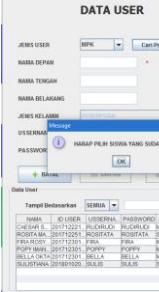
Gambar 4. 16 form mengelola user

### 2) Feedback

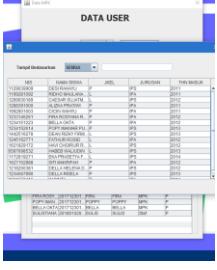
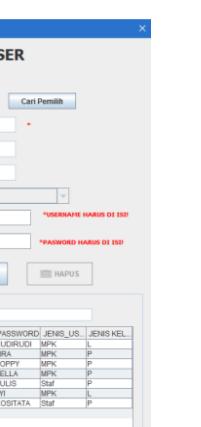
Tabel 4. 32 Pengujian black box mengelola user

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Menambah data dengan memilih jenis user staf dan Memasukkan data staf dengan benar kemudian klik simpan	<p>Test Case: Menambah data user dengan memilih jenis user staf dan Memasukkan data staf dengan benar kemudian klik simpan</p> 	Sistem sukses menyimpan data user dan menampilkan data user		Valid

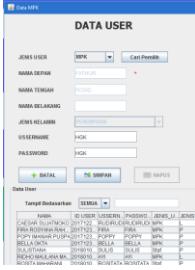
Tabel 4. 32 Pengujian *black box* mengelola user (Lanjutan)

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
2	Menambah data dengan memilih jenis user staf dan Tidak Memasukkan data staf dengan benar kemudian klik simpan		Sistem sukses memunculkan peringatan data yang harus diisi		Valid
3	Menambah data dengan memilih jenis user MPK dan mencari data dari pemilih dengan benar kemudian	 	Sistem sukses mengambil data user dari data pemilih dan menampilkan ke <i>form</i> user		Valid
4	Menambah data dengan memilih jenis user MPK dan mencari data dari pemilih dengan tidak benar (menutup <i>form</i> cari)	 	Sistem memunculkan peringatan “harap pilih siswa yang telah tersedia”		Valid

Tabel 4. 32 Pengujian *black box* mengelola user (Lanjutan)

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
5	Menambah data dengan memilih jenis user MPK dan mencari data dari pemilih dengan tidak benar (tidak double klik)		Sistem memunculkan peringatan “double klik untuk memilih”	 *DOUBLE CLICK UNTUK MENULIS.	Valid
6	Menambah data dengan memilih jenis user MPK dan Tidak Memasukkan data MPK dengan benar kemudian klik simpan		Sistem memunculkan peringatan data yang harus diisi		Valid

Tabel 4. 32 Pengujian *black box* mengelola user (Lanjutan)

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
7	Menambah data dengan memilih jenis user MPK dan memasukkan data MPK yang sudah ada kemudian klik simpan		Sistem memunculkan peringatan “data sudah ada”		Valid
8	Menambah data dengan memilih jenis user MPK dan memasukkan data MPK dengan benar dan kemudian klik simpan		Sistem sukses menyimpan data user dan menampilkan data user		Valid

#### 4.4.5 Iterasi 5

Pada iterasi 5, dilakukan pembuatan proses login dan logout pada sisi administrator. Berikut adalah tahapan *prototype* pada pembuatan prosesnya.

##### a. Communication

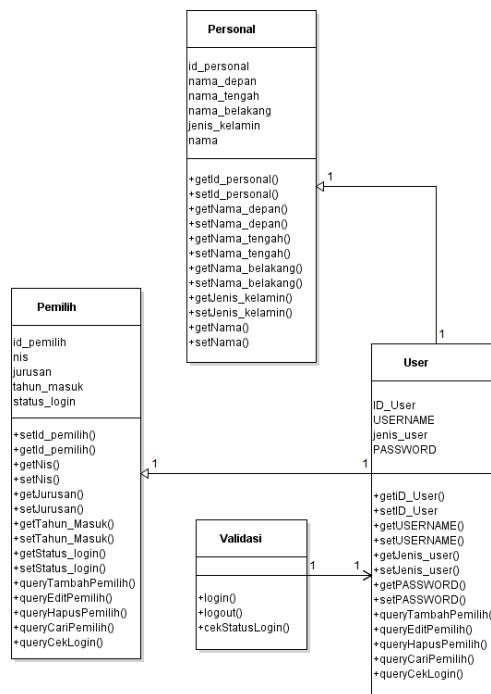
Pada tahap Communication, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi administrator. Use Case yang dilakukan adalah login dan logout.

### b. Quick Plan

Pada tahap *Quick Plan*, macam-macam *usecase* yang digunakan adalah :

- 1) Login : Skenario *usecase* yang digunakan mengacu pada tabel 4.20.
- 2) Logout : Skenario *usecase* yang digunakan mengacu pada tabel 4.21.

*Class diagram* yang digunakan mengacu pada gambar 4.17



Gambar 4. 17 *Class diagram* login dan logout

### c. Quick Design

Pada tahap ini membangun *mockup* dan design database. Design *mockup* yang dibangun dapat dilihat pada gambar 4.18



Gambar 4. 18 Class diagram login

Design database yang dibangun menggunakan tabel user yang mencocokkan dengan username dan password yang sudah didaftarkan oleh administrator pada database.

#### d. Construction of Prototype

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun.

```

1.     public boolean ValidasiLogin(String USERNAME,
2.     String PASSWORD) {
3.         try {
4.             pst = mysql.getConection().prepareStatement ("SELECT
5.                 COUNT(*) AS NUM FROM USER WHERE USER.USERNAME=?"
6.                 AND USER.PASSWORD=?");
7.             pst.setString(1, USERNAME);
8.             pst.setString(2,PASSWORD);
9.             ResultSet rs = pst.executeQuery();
10.            rs.next();
11.            int rowAffected = rs.getInt("NUM");
12.            if (rowAffected == 1) {
13.                return true;
14.            } else {
15.                System.out.println("Invalid username or password!");
16.                return false;
17.            }
18.        } catch (Exception e) {
19.            System.err.println("Error SQL: " + e.getMessage());
20.            return false;
21.        }
22.    }

```

Kode Program 4.7 kode login user

Fungsi ValidasiLogin(String USERNAME, String PASSWORD) untuk mencocokkan pada table user adakah username dan password yang dimaksud, String USERNAME, String PASSWORD adalah parameter yang di ambil dari *form* login , jika benar makan nilai true jika salah false yang kemudian dikirimkan kembali pada *form* login

```

1.     private void btn_okActionPerformed
2.         (java.awt.event.ActionEvent evt) {
3.             // TODO add your handling code here:
4.             String USERNAME = txtusername.getText().toUpperCase();
5.             String PASSWORD = new String (txtpassword.getText())
6.                 .toUpperCase();
7.             CUser usr = new CUser();
8.             boolean cek = usr.ValidasiLogin(USERNAME, PASSWORD);
9.             if ((USERNAME).equals("RUDIRUDI") && cek ){//&&
10.                 // fevoting.restoreMenu(USERNAME);
11.                 fevoting.label_login.setText(USERNAME);
12.                 fevoting.menuitem_login.setText("Logout");
13.                 fevoting.baru=false;
14.                 JOptionPane.showMessageDialog(rootPane, "SELAMAT
15.                     DATANG:"+USERNAME);
15.                 fevoting.masuk_mpk_admin();
16.                 setVisible(false);
17.                 dispose();
18.             }else if((USERNAME).equals("ROSITATA") && cek) {
19.                 // fevoting.restoreMenu(USERNAME);
20.                 fevoting.label_login.setText(USERNAME);
21.                 fevoting.menuitem_login.setText("Logout");
22.                 JOptionPane.showMessageDialog(rootPane, "SELAMAT
23.                     DATANG:"+USERNAME);
24.                 fevoting.baru=false;
25.                 fevoting.masuk_staf();
26.                 setVisible(false);
27.                 dispose();
28.             }else if ((USERNAME).equals("BELLA") && cek ){//&&
29.                 // fevoting.restoreMenu(USERNAME);
30.                 fevoting.label_login.setText(USERNAME);
31.                 fevoting.menuitem_login.setText("Logout");
32.                 fevoting.baru=false;
33.                 JOptionPane.showMessageDialog(rootPane, "SELAMAT
34.                     DATANG:"+USERNAME);
35.                 fevoting.masuk_mpk_tps();
36.                 setVisible(false);
37.                 dispose();
38.             }else if ((USERNAME).equals("FIRA") && cek ){//&&
39.                 // fevoting.restoreMenu(USERNAME);
40.                 fevoting.label_login.setText(USERNAME);
41.                 fevoting.menuitem_login.setText("Logout");
42.                 fevoting.baru=false;
43.                 JOptionPane.showMessageDialog(rootPane, "SELAMAT
44.                     DATANG:"+USERNAME);
45.             }

```

```

48.         fevoting.masuk_mpk_tps();
49.         setVisible(false);
50.         dispose();
51.     }else if ((USERNAME).equals("POPY") && cek ){//&&
52.         // fevoting.restoreMenu(USERNAME);
53.         fevoting.label_login.setText(USERNAME);
54.         fevoting.menuitem_login.setText("Logout");
55.         JOptionPane.showMessageDialog(rootPane,"SELAMAT
56.             DATANG:"+USERNAME);
57.         fevoting.baru=false;
58.         fevoting.masuk_mpk_tps();
59.         setVisible(false);
60.         dispose();
61.     }else {
62.         JOptionPane.showMessageDialog(rootPane,"Silahkan
63.             periksa kembali password dan username anda");
64.     }

```

Kode Program 4.8 kode hak akses login user

Fungsi btn\_okActionPerformed sebagai untuk mengatur hak akses sesuai username yang telah di cocokan pada database tabel user

#### e. Deployment Delivery and Feedback

Pada tahap ini merupakan tahapan pengujian pada iterasi login dan logout user, apabila pengguna tidak sesuai dengan hasil yang di harapkan maka pengembang memperbaiki sistem tersebut sehingga aplikasi atau sistem yang di buat sesuai dengan yang di inginkan oleh pengguna pengujian ini menggunakan *black box* pada gambar 4.19 , gambar 4.20, gambar 4.21, gambar 4.22, gambar 4.23 adalah Deployment Delivery dan pada tabel 4.33 adalah pengujian *black box* validasi user



Gambar 4. 19 Activity menu sebelum login



Gambar 4. 20 Activity menu sesudah *login MPK (administrator)*



Gambar 4. 22 Activity menu sesudah *login MPK (panitia tps)*



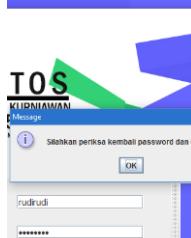
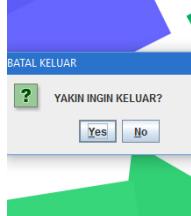
Gambar 4. 23 Activity menu sesudah *login Staf* Kesiswaan



Gambar 4. 24 Activity menu sesudah *logout*

## 2) Feedback

Tabel 4. 33 Pengujian *black box* validasi user

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Memasukkan data username dan password dengan benar kemudian klik OK		Data yang dimasukkan benar dan menampilkan halaman menu sesuai hak akses		Valid
2	Tidak memasukkan data username dan password dengan benar kemudian klik OK		Kembali ke form login dan menampilkan pemberitahuan “Silahkan periksa kembali username dan password anda”		Valid
3	Menekan button logout		Menampilkan dialog dengan pesan “apakah yakin logout? ”		Valid

#### 4.4.6 Iterasi 6

Pada iterasi 6, dilakukan pembuatan proses cetak rekapitulasi hasil suara pada sisi administrator (Staf Kesiswaan). Berikut adalah tahapan *prototype* pada pembuatan proses proses cetak rekapitulasi hasil suara

##### a. *Communication*

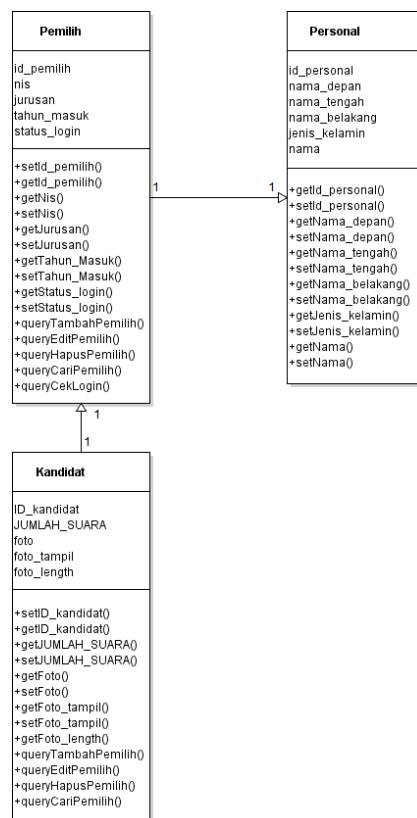
Pada tahap *Communication*, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi administrator (Staf kesiswaan). Use Case yang dilakukan adalah mencetak rekapitulasi suara

##### b. *Quick Plan*

Pada tahap *Quick Plan*, macam-macam *usecase* yang digunakan adalah:

- 1). Cetak rekapitulasi: Skenario *usecase* mengacu pada tabel 4.22.

*Class diagram* yang digunakan mengacu pada gambar 4.24

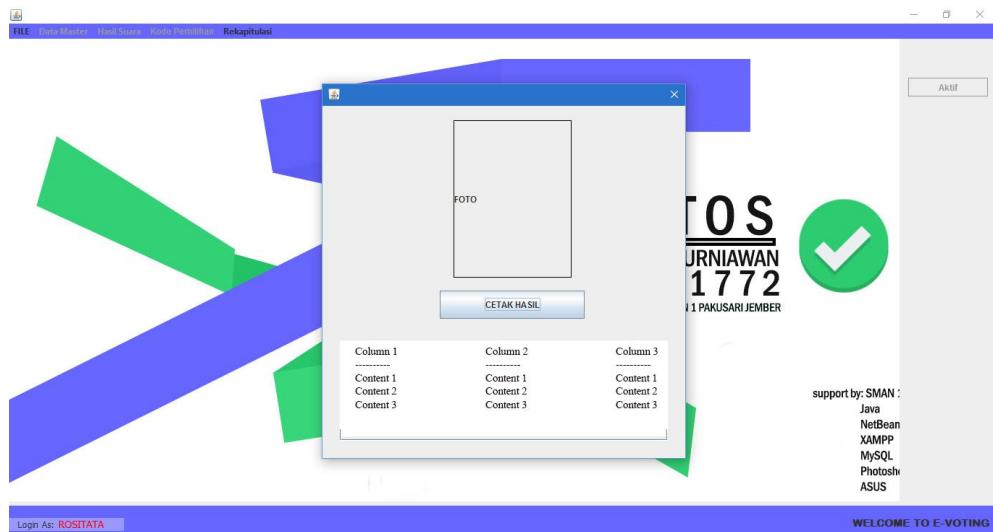


Gambar 4. 24 *Class diagram* cetak rekapitulasi

### c. Quick Design

Pada tahap ini membangun *mockup* dan design database.

Design *mockup* yang dibangun dapat dilihat pada gambar 4.25



Gambar 4. 25 Design *mockup* cetak rekapitulasi suara

Design database yang dibangun adalah tabel kandidat mengacu pada tabel 4.34.

Tabel 4. 34 Tabel kandidat

No	Nama Field	Jenis	Deskripsi
1	ID_KANDIDAT	varchar(17)	<i>Primary key</i>
2	ID PEMILIH	varchar(17)	<i>Foreign key</i>
3	FOTO	Blob	
4	JUMLAH_SUARA	int(3)	

### d. Construction of Prototype

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun.

```

1.    private void btn_cetakActionPerformed
2.        (java.awt.event.ActionEvent evt) {
3.            // TODO add your handling code here:
4.            try{
5.                String NamaFile =
6.                "./src/evotingserver/laporan_hasil_suara.jasper";
7.                Class.forName("com.mysql.jdbc.Driver").newInstance();
8.                Connection koneksi = DriverManager.getConnection
9.                    ("jdbc:mysql://localhost/EVOTING","root","");
10.               HashMap param = new HashMap();

```

```

11.         param.clear();
12.         JasperPrint JPrint = JasperFillManager.fillReport
13.             (NamaFile, param, koneksi);
14.             JasperViewer.viewReport(JPrint);
15.         }catch(Exception ex){
16.             System.out.println(ex);
17.         }
18.     }

```

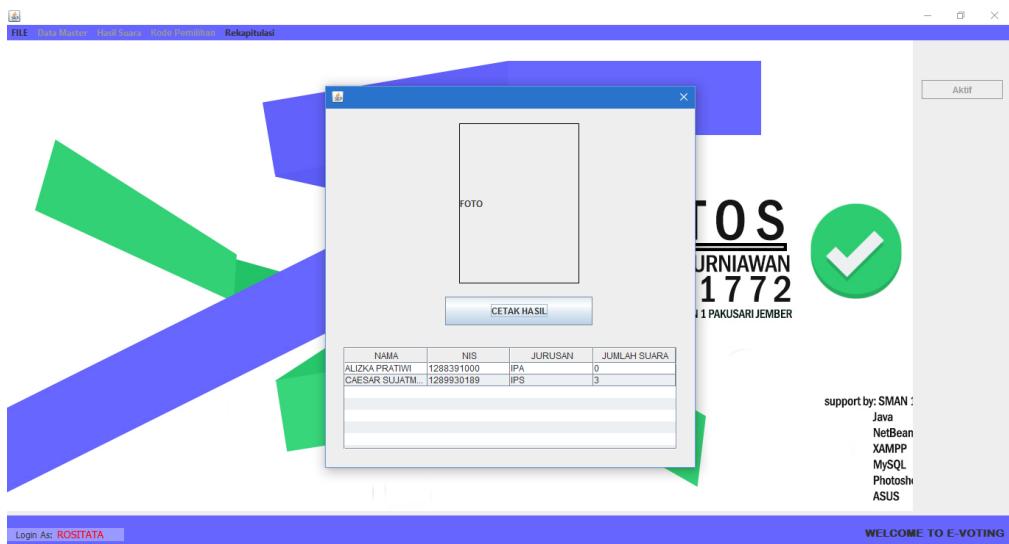
Kode Program 4.9 kode cetak rekapitulasi hasil suara

Fungsi void btn\_cetakActionPerformed (java.awt.event.ActionEvent evt) diatas adalah untuk mencetak kode login dengan menggunakan parameter nis untuk mengambil data pemilih.

#### e. Deployment Delivery and Feedback

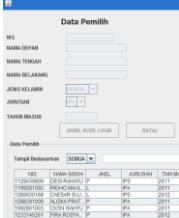
Pada tahap ini merupakan tahapan pengujian pada iterasi mencetak hasil rekapitulasi suara, apabila pengguna tidak sesuai dengan hasil yang di harapkan maka pengembang memperbaiki sistem tersebut sehingga aplikasi atau sistem yang di buat sesuai dengan yang di inginkan oleh pengguna pengujian ini menggunakan *black box* pada gambar 4.40 adalah Deployment Delivery dan pada tabel 4.35 adalah pengujian *black box* mencetak hasil rekapitulasi suara

##### 1) Deployment delivery



Gambar 4. 26 form cetak rekapitulasi suara

Tabel 4. 35 Pengujian black box cetak rekapitulasi

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Mencetak kode pemilih dengan mengeklik data pemilih yang ingin dicetak kemudian pilih <i>button</i> ambil kode login	 	Sistem sukses menyetak kode sesuai data pemilih		Valid

#### 4.4.7 Iterasi 7

Pada iterasi 7, dilakukan pembuatan proses login kode pemilihan dari sisi pengguna(pemilih). Berikut adalah tahapan *prototype* pada pembuatan proses proses login kode pemilihan

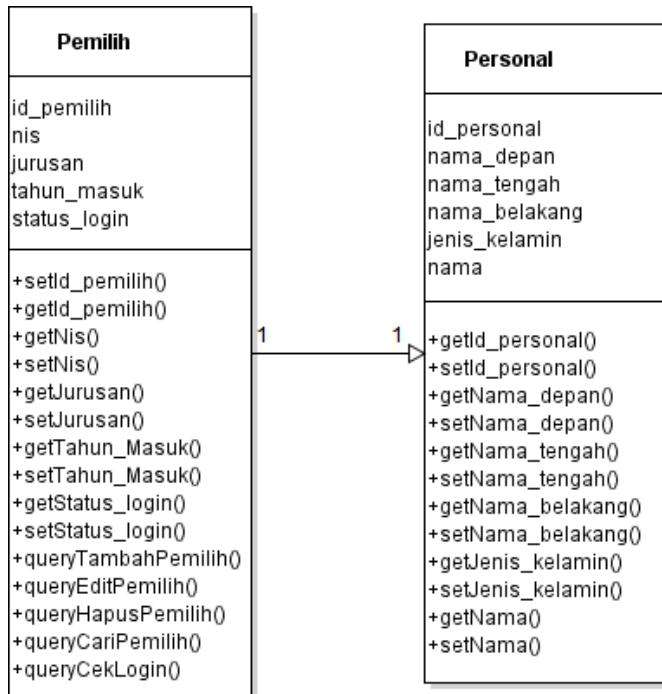
##### a. *Communication*

Pada tahap *Communication*, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi pengguna(pemilih). Use Case yang dilakukan adalah login kode pemilihan

##### b. *Quick Plan*

Pada tahap *Quick Plan*, macam-macam *usecase* yang digunakan adalah:

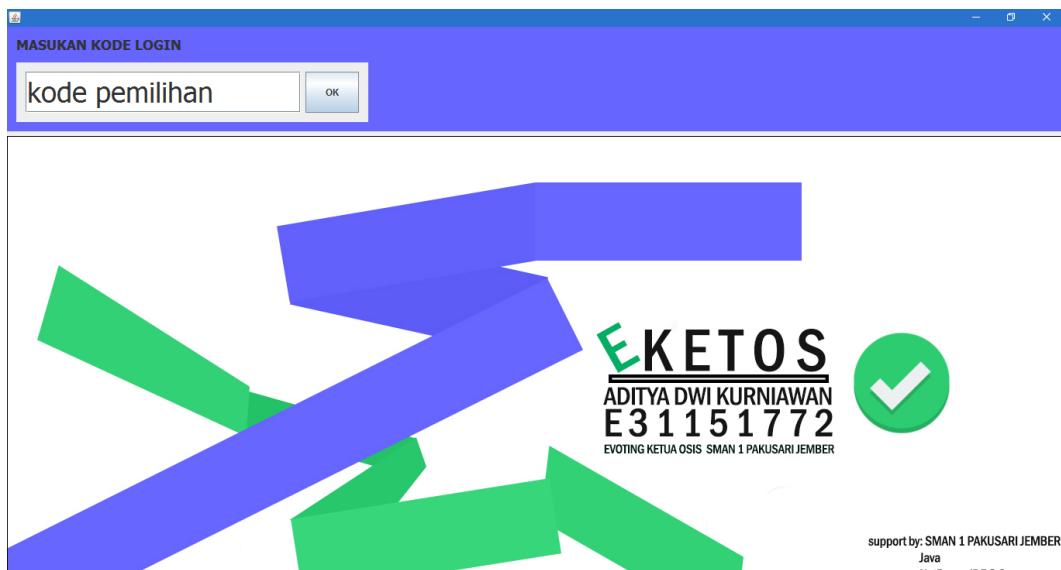
- 1) login kode pemilihan: Skenario *usecase* mengacu pada tabel 4.27.



Gambar 4. 27 *Class diagram* login pemilih

### c. Quick Design

Pada tahap ini membangun *mockup* dan design database. Design *mockup* yang dibangun dapat dilihat pada gambar 4.28



Gambar 4. 28 Design *mockup* login pemilih

#### d. Construction of Prototype

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun.

```

1.     public boolean ValidasiLogin
2.           (String ID PEMILIH ,Integer CEK_LOGIN) {
3.               try {
4.                   pst = mysql.getConection().prepareStatement
5.                           ("SELECT COUNT(*) AS NUM FROM PEMILIH
6.                               WHERE PEMILIH.ID PEMILIH=?AND
7.                                   PEMILIH.STATUS LOGIN=?");
8.                   pst.setString(1, ID PEMILIH);
9.                   pst.setInt(2, CEK_LOGIN);
10.                  ResultSet rs = pst.executeQuery();
11.                  System.out.println(pst);
12.                  rs.next();
13.                  int rowAffected = rs.getInt("NUM");
14.                  if (rowAffected == 1 ) {
15.                      return true;
16.                  } else {
17.                      System.out.println("ID SUDAH DIGUNAKAN ATAU
18.                          TIDAK VALID!");
19.                      return false;
20.                  }
21.              }catch (Exception e) {
22.                  System.err.println("Error SQL Validasi Login
23.                      Pemilih Di QUERY: " + e.getMessage());
24.                  return false;
25.              }
26.          }
27.          public int edit(Pemilih pem){
28.              try{
29.                  st = mysql.getConection().createStatement();
30.                  String SQL;
31.                  SQL = "UPDATE PEMILIH SET "+" STATUS_LOGIN=" +""+
32.                      pem.getStatus_login()+" "+ WHERE ID PEMILIH"+
33.                      "=" + pem.getId_pemilih()+"";
34.                  st.addBatch(SQL);
35.                  int[] count = st.executeBatch();
36.                  return count.length;
37.              } catch (SQLException e) {
38.                  System.err.println("Error UPDATE STATUS LOGIN
39.                      Kandidat Di Query! " + e.getMessage());
40.                  return 0;
41.              }
42.          }

```

Kode Program 4.10 kode model login pemilih

Fungsi ValidasiLogin(String ID PEMILIH ,Integer CEK\_LOGIN) untuk mencocokkan pada database atau table user apakah ada username dan password yang yang dimaksud, String ID PEMILIH ,Integer CEK\_LOGIN adalah parameter yang dimana cek\_login bernilai 1 untuk masuk ke *form* pemilihan dan

ID PEMILIH di ambil dari *form* masukan kode, jika benar makan nilai true jika salah false, yang kemudian dikirimkan kembali pada *form* pemilihan

Fungi public int edit(Pemilih pem) untuk mengubah status bahwa pemilih sudah memasukkan kode ke *form* pemilihan, status\_login pada tabel pemilih bernilai 0 yang artinya pemilih tidak bisa memasukkan kode pemilihan untuk kedua kalinya

```

1.     private void btn_okActionPerformed
2.         (java.awt.event.ActionEvent evt) {
3.             // TODO add your handling code here:
4.             try{
5.                 String ID PEMILIH SUARA =
6.                 txt_id_pemilih.getText().toUpperCase();
7.                 Integer CEK_LOGIN =(CEK_STATUS_LOGIN);
8.                 CPemilih cpemilih = new CPemilih();
9.                 boolean cek = cpemilih.ValidasiLogin
10.                     (ID PEMILIH SUARA,CEK_LOGIN);
11.                     if ( cek==true){
12.                         JOptionPane.showMessageDialog(rootPane,"SELAMAT
13.                                         DATANG");
14.                         JFrame frame = (JFrame) SwingUtilities
15.                                         .getWindowAncestor(this);
16.                         FormPemilihan formpemilihan = new
17.                             FormPemilihan(frame,true);
18.                             formpemilihan.setLocationRelativeTo(null);
19.                             formpemilihan.setParent(this);
20.                             formpemilihan.setVisible(true);
21.                             Pemilih pemilih = new Pemilih();
22.                             pemilih.setId_pemilih(ID PEMILIH SUARA);
23.                             pemilih.setStatus_login(STATUS_LOGIN);
24.                             boolean simpan = false;
25.                             if (cpemilih.edit(pemilih) > 0) {
26.                                 if (cpemilih.save()) {
27.                                     simpan = true;
28.                                 }
29.                             }
30.                             setComponentsDisplay(COMPONENTS DISPLAY.UTAMA);
31.                         }else {
32.                             JOptionPane.showMessageDialog(rootPane,"NO LOGIN
33.                                         TIDAK VALID,"+"ATAU
34.                                         SUDAH DIGUNAKAN");
35.                             txt_id_pemilih.setFocusable(true);
36.                         }
37.                     }catch(Exception ex){
38.                         System.out.println(ex);
39.                     }
40.             }

```

Kode Program 4.11 kode login pemilih

Fungsi btn\_okActionPerformed mengambil parameter dari *form* tampilan ke model pemilih yang nantinya diproses ke database

`txt_id_pemilih.getText()` digunakan sebagai pengambilan kode pemilihan pada *form* pemilih untuk nantinya dijadikan sebagai parameter

`Integer CEK_LOGIN =(CEK_STATUS_LOGIN)` digunakan untuk memvalidasi dimana `CEK_STATUS_LOGIN` bernilai 1

`if ( cek==true){` digunakan untuk mengecek nilai yang telah dikirimkan model,jika nilai true akan muncul halaman pemilihan kemudian `STATUS_LOGIN` pada tabel pemilih diubah `pemilih.setStatus_login(STATUS_LOGIN)` yang dimana nilai `STATUS_LOGIN` bernilai 0 yang berparameterkan `pemilih.setId_pemilih (ID PEMILIH SUARA)`. Jika nilai bernilai false akan memunculkan notifikasi “NO LOGIN TIDAK VALID ATAU SUDAH DIGUNAKAN”

#### e. Deployment Delivery and Feedback

Pada tahap ini merupakan tahapan pengujian pada iterasi login Pemilih, apabila pengguna tidak sesuai dengan hasil yang di harapkan maka pengembang memperbaiki sistem tersebut sehingga aplikasi atau sistem yang di buat sesuai dengan yang di inginkan oleh pengguna pengujian ini menggunakan *black box* pada gambar 4.30 adalah Deployment Delivery dan pada tabel 4.36 adalah pengujian *black box* login pemilih.

1) Deployment delivery



Gambar 4. 30 Design login pemilih

Tabel 4. 36 Pengujian black box login pemilih

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Memasukkan kode pemilihan dengan benar kemudian klik OK		Data yang dimasukkan benar dan menampilkan dialog "selamat datang" dan halaman pemilihan		Valid

Tabel 4. 36 Pengujian black box login pemilih (Lanjutan)

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
2	Tidak memasukkan kode pemilihan dengan benar kemudian klik OK		Kembali ke <i>form login</i> dan menampilkan pemberitahuan “No login tidak valid, atau sudah digunakan”		Valid
3	memasukkan kode pemilihan dengan benar, namun sudah memilih kemudian klik OK		Kembali ke <i>form login</i> dan menampilkan pemberitahuan “No login tidak valid, atau sudah digunakan”		Valid

#### 4.4.8 Iterasi 8

Pada Iterasi 8, dilakukan pembuatan proses pemilihan kandidat dari sisi pemilih.. Berikut adalah tahapan *prototype* pada pembuatan proses pemilihan kandidat

##### a. *Communication*

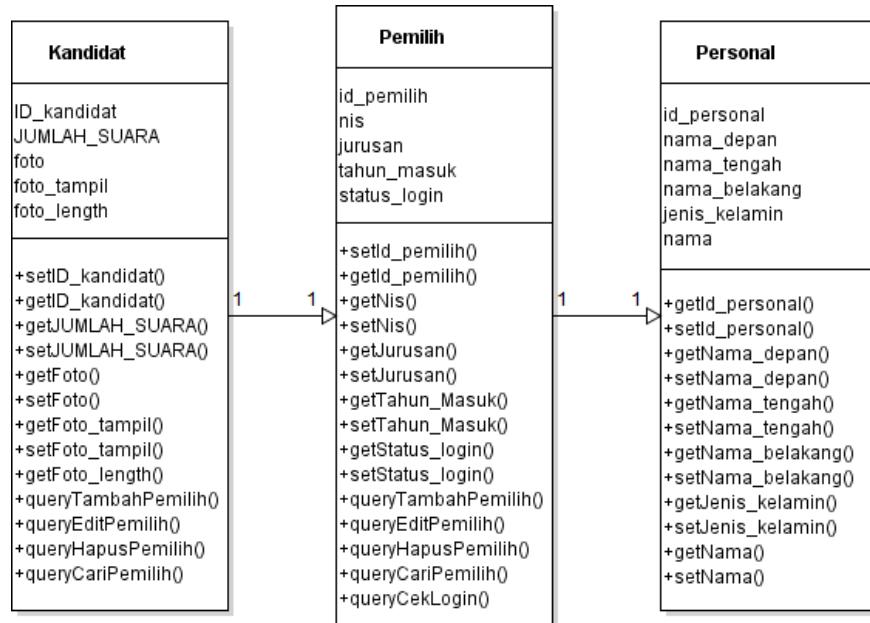
Pada tahap Communication, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi pemilih. *Use Case* yang dilakukan adalah memilih kandidat, melihat 4.25

### b. Quick Plan

Pada tahap *Quick Plan*, macam-macam *usecase* yang digunakan adalah:

- 1) Memilih kandidat : Skenario *usecase* mengacu pada tabel 4.14.
- 2) Melihat kandidat : Skenario *usecase* mengacu pada tabel 4.25.

*Class diagram* yang digunakan mengacu pada gambar 4.31

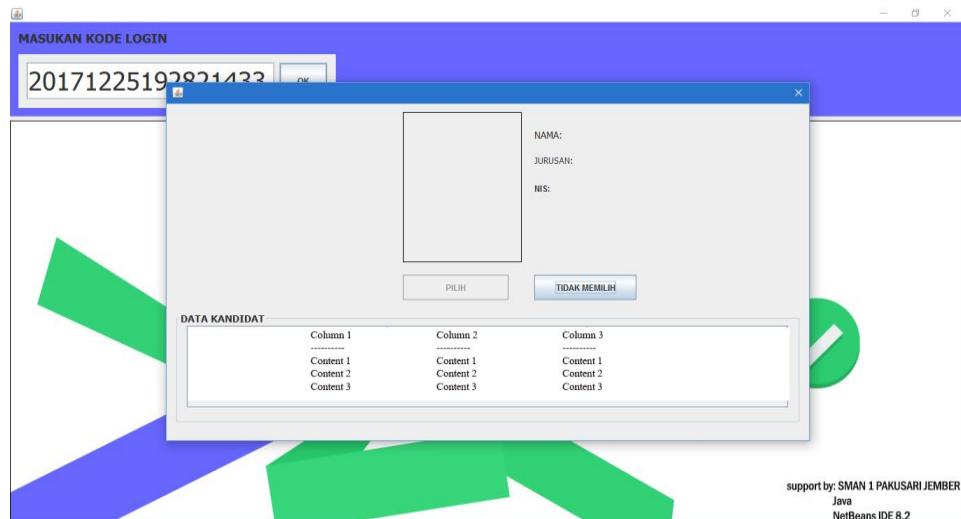


. Gambar 4. 31 *Class diagram* kandidat

### c. Quick Design

Pada tahap ini membangun *mockup* dan design database.

Design *mockup* yang dibangun dapat dilihat pada gambar 4.32



Gambar 4. 32 Design *mockup* pemilihan kandidat

Design database yang dibangun adalah tabel kandidat mengacu pada tabel 4.36.

Tabel 4. 37 Tabel kandidat

No	Nama Field	Jenis	Deskripsi
1	ID_KANDIDAT	varchar(17)	<i>Primary key</i>
2	ID PEMILIH	varchar(17)	<i>Foreign key</i>
3	FOTO	Blob	
4	JUMLAH_SUARA	int(3)	

#### d. *Construction of Prototype*

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun.

```

1.   public ResultSet show(KEYWORDS keyword, String key) {
2.       try {
3.           String sql = "SELECT * FROM V_KANDIDAT";
4.           System.out.println(sql);
5.           pst = mysql.getConection().prepareStatement(sql);
6.           return(pst.executeQuery());
7.       } catch(java.sql.SQLException e) {
8.           System.err.println("Error SQL v_kandidat: " +
9.                           e.getMessage());
10.          return null;
11.      }
12.  }
13.  public int pilih(Kandidat kan) {

```

```

14.         try{
15.             st = mysql.getConection().createStatement();
16.             String SQL;
17.             SQL = "UPDATE KANDIDAT SET "+
18.                 " JUMLAH_SUARA=JUMLAH_SUARA+" +""+
19.                     kan.getJUMLAH_SUARA()+" "+"
20.                     " WHERE id_kandidat"+ "=" +
21.                         kan.getID_kandidat()+"";
22.             st.addBatch(SQL);
23.             int[] count = st.executeBatch();
24.             return count.length;
25.         } catch (SQLException e) {
26.             System.err.println("Error Memilih Kandidat Di
27.                                 Query! " + e.getMessage());
28.             return 0;
29.         }
30.     }
31.     public boolean save() {
32.         try {
33.             mysql.getConection().commit();
34.             return true;
35.         } catch (SQLException e) {
36.             try {
37.                 mysql.getConection().rollback();
38.             } catch (SQLException e1) {
39.                 System.err.println("Error: " + e1.getMessage());
40.             }
41.             System.err.println("Error: " + e.getMessage());
42.             return false;
43.         }
44.     }

```

#### Kode Program 4.12 model pemilihan kandidat

Fungsi `ResultSet show (KEYWORDS keyword, String key)` yang digunakan untuk menampilkan seluruh data dari tabel kandidat

Fungsi `pilih(Kandidat kan)` digunakan untuk menyimpan data masukan dari FormPemilihan yang dimasukkan ke dalam array kemudian disimpan dalam variabel `st`. `kan.getID_kandidatl` digunakan sebagai id acuan dalam mengubah data kandidat.

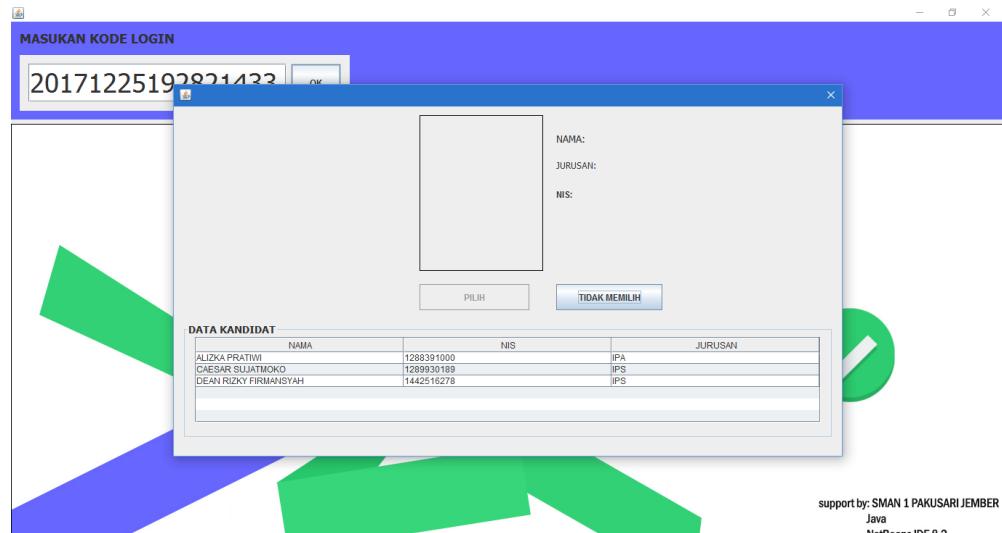
Fungsi `save` digunakan untuk pemanggilan koneksi MySQL yang digunakan untuk menyimpan data masukan dari FormPemilihan.

#### e. Deployment Delivery and Feedback

Pada tahap ini merupakan tahapan pengujian pada iterasi pemilihan kandidat apabila pengguna tidak sesuai dengan hasil yang di harapkan maka pengembang memperbaiki sistem tersebut sehingga aplikasi atau sistem yang di

buat sesuai dengan yang di inginkan oleh pengguna pengujian ini menggunakan *black box* pada gambar 4.46 adalah Deployment Delivery dan pada tabel 4.38 adalah pengujian *black box* pemilihan kandidat.

### 1) Deployment delivery



Gambar 4. 33 form pemilihan kandidat

### 2) Feedback

Tabel 4. 38 Pengujian black box memilih kandidat

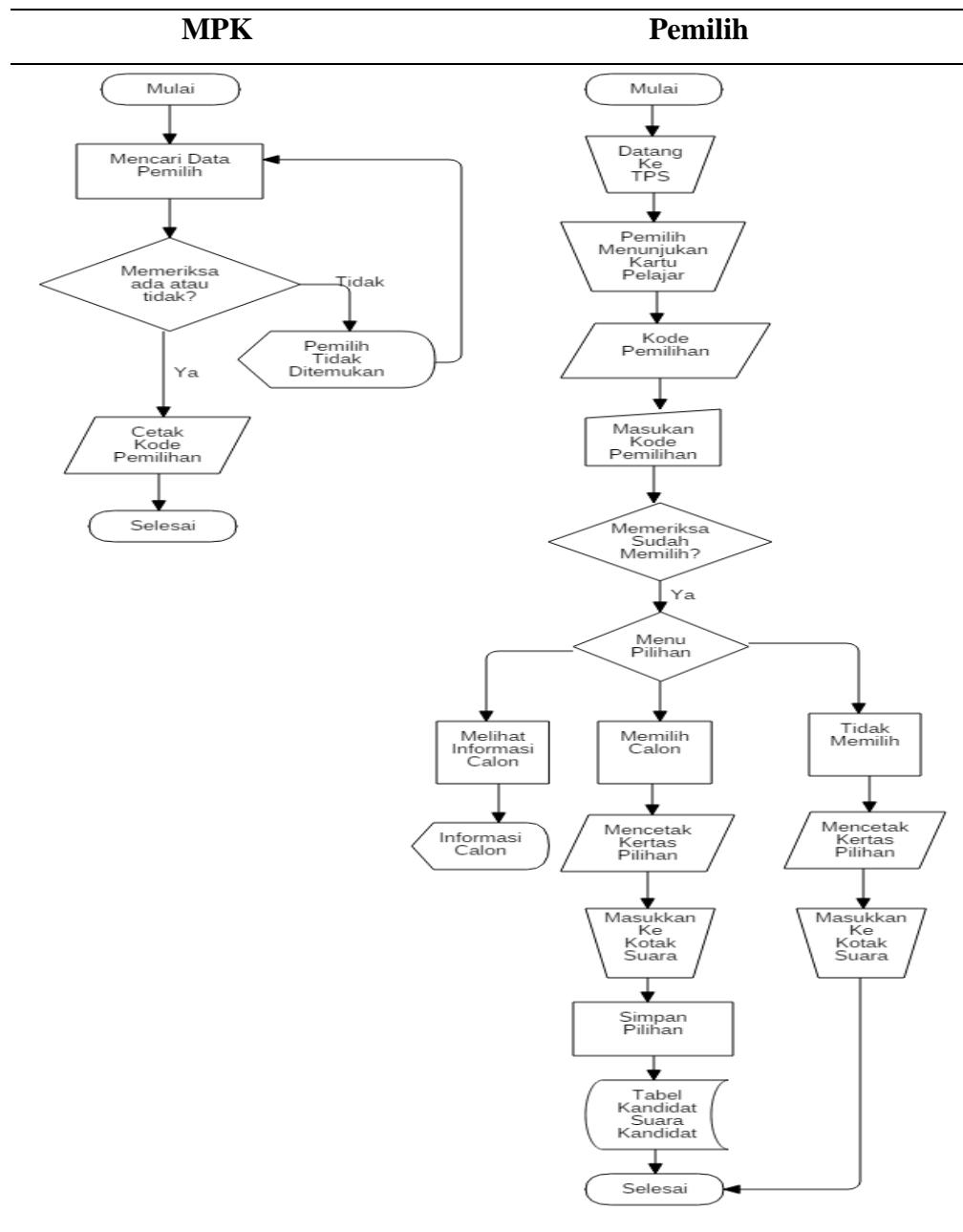
No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Memilih kandidat dengan benar kemudian menekan button pilih		Menampilkan dialog "Apakah anda yakin melilih CAESAR" dengan input nama "kandidat"		Valid

Tabel 4. 38 Pengujian black box memilih kandidat (Kandidat)

No	Skenario	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
2	Jika memilih yes		Data yang pilih benar dan menampilkan dialog "anda telah memilih" lalu otomatis mencetak pilihan yang dipilih		Valid
3	Jika memilih no		Kembali ke form pemilihan		Valid

#### 4.5 Flowcart Sistem Pemilihan E-voting Dan Manual

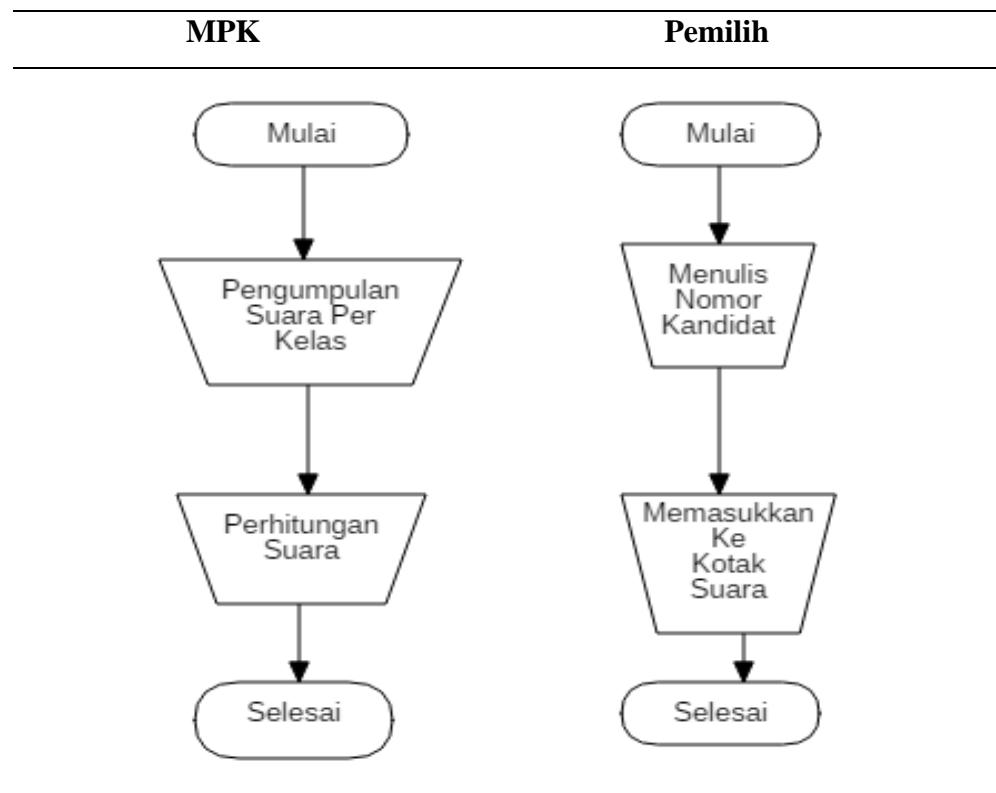
Tabel 4. 39 *Flowchart* Sistem Pemilihan E-voting



Berdasarkan tabel 4.39: Disini MPK (majelis perwakilan kelas) yang sebagai penitip TPS tidak perlu lagi keliling ke tiap-tiap kelas, MPK melakukan pencarian data pemilih berdasarkan NIS atau nama yang nantinya untuk mencetak kode login pemilih di masing-masing TPS, Dan pemilih wajib menunjukan kartu pelajar

sebagai syarat mengambil kode pemilih ke panitia TPS (MPK). Setelah selesai memilih, pemilih memasukkan kertas suara sebagai pengamanan untuk perhitungan ulang secara manual jika diperlukan

Tabel 4. 40 *Flowchart* Sistem Pemilihan Manual



Berdasarkan tabel 4.40: Disini MPK (majelis perwakilan kelas) keliling ke tiap-tiap kelas untuk mengumpulkan suara, Pemilih Harus menunggu MPK datang ke kelas masing-masing untuk memilih sehingga membutuhkan waktu yang banyak dan mengganggu waktu pembelajaran siswa. Pada waktu pemilihan pemilih diberi kertas yang digunakan untuk menuliskan nomor calon kandidat yang kemudian dimasukkan ke kotak suara.

## **BAB 5. KESIMPULAN DAN SARAN**

### **5.1 Kesimpulan**

Berdasarkan hasil pembahasan Sistem Informasi E-voting pemilihan ketua OSIS SMAN 1 Pakusari Jember didapatkan kesimpulan sebagai berikut :

- a. Sistem informasi ini dapat membantu siswa dalam pemilihan ketua OSIS di SMAN 1 PAKUSARI JEMBER. Kemudahan yang didapat dari siswa adalah kemudahan pada saat pemilihan karena tidak sampai mengganggu jam pelajaran
- b. Sistem informasi ini dapat membantu panitia dalam pemilihan ketua OSIS di SMAN 1 PAKUSARI JEMBER. Kemudahan yang didapat dari panitia adalah tidak mengeluarkan tenaga untuk keliling ke tiap - tiap kelas, karena telah disediakan beberapa TPS, dan pada saat perhitungan suara, hasil akurat karena dihitung oleh sistem dan manual

### **5.2 Saran**

Berdasarkan kesimpulan-kesimpulan yang telah dikemukakan. Dapat diajukan beberapa saran untuk pengembangan lebih lanjut :

- a. Membuat desain lebih bersifat *user interface*.
- b. Menambahkan fitur-fitur yang lain agar lebih lengkap dan nyaman digunakan
- c. Menambahkan fitur keamanan jaringan pada saat pemilihan suara

## DAFTAR PUSTAKA

Darmawan, Ikhsab., Nurhanjati, Nurhajati., Kartini, Evida..2014.*memahami E-voting Berkaca dari Pengalaman Negara-negara lain dan jembrana (Bali)*. Diambil dari:

[https://books.google.co.id/books?id=KEFODAAAQBAJ&pg=PA96&lpg=PA96&dq=voting+adalah+menurut&source=bl&ots=JBCXVjnh3f&sig=bpfOcxIwz1LWeYK7VzpLFBu4\\_A&hl=id&sa=X&ved=0ahUKEwizuInY2cHUAhVB9WMKHUFyCb04ChDoAQgSMAM#v=snippet&q=voting%20adalah&f=false](https://books.google.co.id/books?id=KEFODAAAQBAJ&pg=PA96&lpg=PA96&dq=voting+adalah+menurut&source=bl&ots=JBCXVjnh3f&sig=bpfOcxIwz1LWeYK7VzpLFBu4_A&hl=id&sa=X&ved=0ahUKEwizuInY2cHUAhVB9WMKHUFyCb04ChDoAQgSMAM#v=snippet&q=voting%20adalah&f=false) (16 juni 2017)

Fathansyah.2015.*Basisdata*.Palasari 82 Bandung.Informatik

Hanifah, N. 2014. *Analisa Sistem Informasi Campus Service Iduhelp Berbasis Ilearning Pada Perguruan Tinggi Raharja*. Tangerang : Sekolah Tinggi Manajemen dan Ilmu Komputer Raharja.

Hutagalung, K. M. 2012. *PERANCANGAN PERANGKAT E-VOTING BERBASIS E-KTP* , STMIK Triguna Dharma,Medan

Muhid, A.(tanpa tahun). *Pengertian Sistem Informasi*

Diambil dari:

[https://www.academia.edu/10331212/Pengertian\\_Sistem\\_Informasi](https://www.academia.edu/10331212/Pengertian_Sistem_Informasi)  
(18 juni 2017)

Nugraha, E.D. 2015. *Sistem Informasi Pemilihan Ketua Umum Deperma Di Politeknik Negeri Jember Berbasis Web*. Tugas .Akhir Politehnik Negeri Jember

Nugroho, W.A. 2011.*Perancangan E-Voting Berbasis Web ( Studi Kasus Pemilihan Kepala Daerah Sukoharjo)* .Skripsi. Universitas Islam Negeri Sunan Kalijaga Yogyakarta

Prima Rosa, P. H., Priyatma, E., Binanto, I., Polina, A. 2014. MENJAJAKI KEMUNGKINAN PENERAPAN SISTEM E-VOTING UNTUK PEMILU DI INDONESIA: BELAJAR DARI KABUPATEN JEMBRANA. Diambil dari:

[https://www.researchgate.net/profile/Paulina\\_Rosa/publication/294661242\\_MENJAJAKI\\_KEMUNGKINAN\\_PENERAPAN\\_SISTEM\\_EVOTING\\_UNTUK PEMILU\\_DI\\_INDONESIA\\_BELAJAR\\_DARI\\_KABUPATEN\\_JEMBRANA\\_BALI/links/56c2d55f08ae2dc3eb88d97e/MENJAJAKI-KEMUNGKINAN-PENERAPAN-SISTEM-E-VOTING-UNTUK-PEMILU-DI-INDONESIA-BELAJAR-DARI-KABUPATEN-JEMBRANA-BALI.pdf](https://www.researchgate.net/profile/Paulina_Rosa/publication/294661242_MENJAJAKI_KEMUNGKINAN_PENERAPAN_SISTEM_EVOTING_UNTUK PEMILU_DI_INDONESIA_BELAJAR_DARI_KABUPATEN_JEMBRANA_BALI/links/56c2d55f08ae2dc3eb88d97e/MENJAJAKI-KEMUNGKINAN-PENERAPAN-SISTEM-E-VOTING-UNTUK-PEMILU-DI-INDONESIA-BELAJAR-DARI-KABUPATEN-JEMBRANA-BALI.pdf) (29 mei 2017)

Ritonga,P.T *Pengertian Unified Modelling Language UML dan Modelnya*

*Menurut Pakar.* Diambil dari:

<http://www.bangpahmi.com/2015/04/pengertian-unified-modelling-language-uml-dan-modelnya-menurut-pakar.html> (5 maret 2017)

S, Rosa A.dan M. Shalahuddin. 2016. Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung: Informatika

Syamsul, Zein. 2014. *sejarah singkat tentang OSIS.*

Diambil dari: <https://www.scribd.com/doc/228534701/Buku-Panduan-Pelaksanaan-OSIS-dan-MPK>. (12 february 2017)

## **LAMPIRAN**

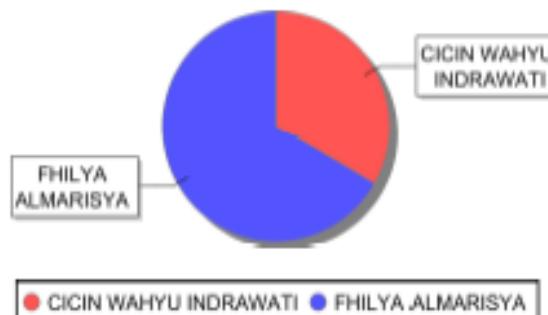
## Lampiran 1. Laporan Hasil Rekapitulasi Suara

**LAPORAN HASIL PEMILIHAN KETUA OSIS  
SMAN 1 PAKUSARI JEMBER**

---

Diamhil: 17/01/2018 7:28

nis	NAMA	JUMLAH
1993001922	CICIN WAHYU INDRAWATI	1
9920037881	FHILYA ALMARISYA	2
	<b>Total Suara</b>	<b>3</b>
	<b>Suara Tidak SAH</b>	<b>2</b>
	<b>Pemilih Sudah Memilih</b>	<b>5</b>
	<b>Pemilih Belum Memilih</b>	<b>17</b>
	<b>Jumlah Pemilih</b>	<b>22</b>



Waka Kணிவன்  
SMAN 1 PAKUSARI.JFMRF.R

M Ali Wafa,S.Pd  
NIP:19830505 199802 1 001

## Lampiran 2. Hasil Cetak Kode Login Pemilih

<b>LOGIN EVOTING</b>			
<i>Sifat:Rahasia</i>			
Dambil: 17/01/2018 7.25 AM			
<b>NIS</b>	<b>NAMA</b>		<b>ID LOGIN</b>
1627388819	DEAN	RIZKY	FIRMANSYA
			20180109234731017