

MANUAL BOOK PROGRAM APLIKASI PEMBAYARAN SPP PADA SMK JAKARTA TIMUR 2

Tim Penyusun :

- 1. Siti Nurajizah**
- 2. Arum Widyastuti**
- 3. Dahlia Sarkawi**
- 4. Anggi Oktaviani**
- 5. Hikmatulloh**
- 6. Safitri Linawati**
- 7. Fathur Rohman**
- 8. Amin Setio Lestinarsih**
- 9. Normah**
- 10. Deny Novianti**
- 11. Ainun Zumarniansyah**
- 12. Ita Yulianti**
- 13. Ami Rahmawati**



**Jakarta
2020**

PENGANTAR

Assalammu'alaikum Warohmatullahi Wabarokaatuh,

Alhamdulillahirabbil'alaamiin

Segala puji dan syukur kepada Allah SWT yang telah melimpahkan rahmat, hidayah serta inayah-Nya sehingga akhirnya penulis dapat menyelesaikan Manual *Book* penggunaan Program Aplikasi Pembayaran SPP pada SMK Jakarta Timur 2.

Penulis menyadari bahwa manual *book* ini masih jauh dari sempurna, sehingga diharapkan kepada para pembaca agar berkenan memberikan kritik dan sarannya yang bersifat membangun untuk perbaikan untuk kedepannya. Semoga panduan ini dapat bermanfaat bagi pembaca maupun penulis. Akhir kata penulis ucapkan terimakasih.

Wassalamualaikum Warohmatullahi Wabarakatuh

Jakarta, November 2020

Tim Penyusun

Tim Penyusun

Materi dan Pengujian:

Siti Nurajizah
Dahlia Sarkawi
Hikmatulloh
Safitri Linawati

Desain dan Animasi:

Fathur Rohman
Ita Yulianti
Normah
Ami Rahmawati

Scripting dan Programming

Arum Widyastuti
Anggi Oktaviani
Amin Setio Lestinarsih
Deny Novianti
Ainun Zumarniansyah

DAFTAR ISI

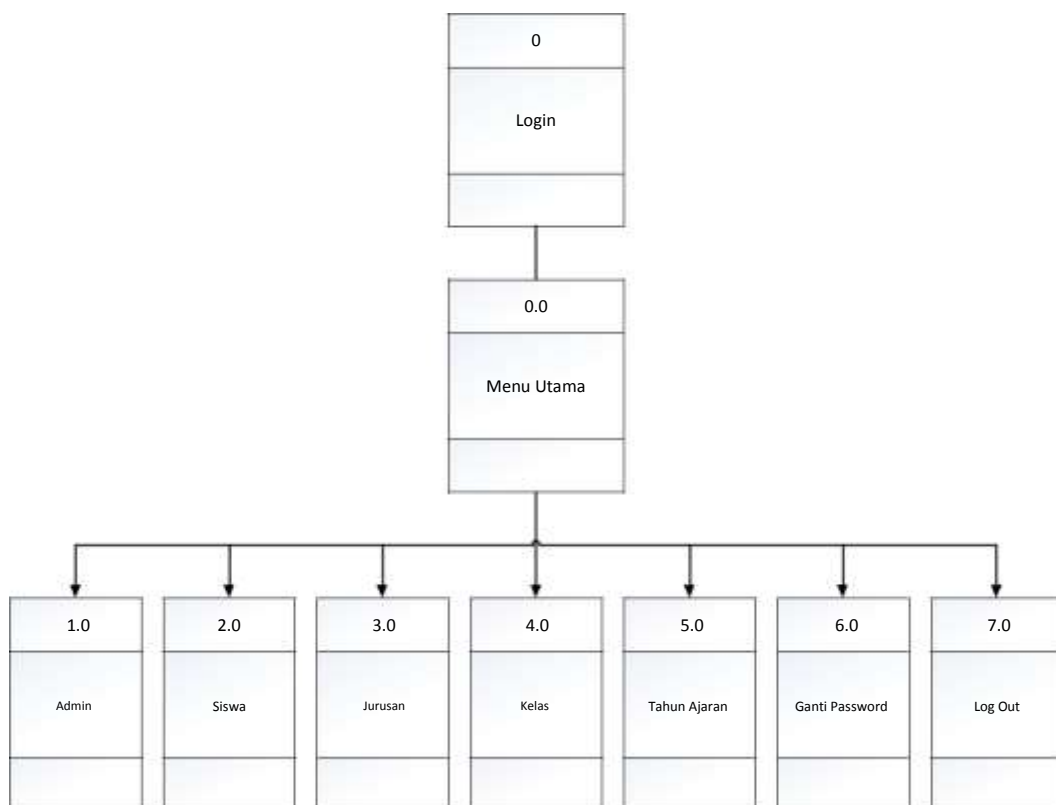
	Halaman
Cover	1
Pengantar.....	2
Tim Penyusun	3
Daftar Isi.....	4
I. Pendahuluan	5
II. HIPO Administrator.....	5
III. HIPO Tata Usaha.....	6
IV. HIPO Kepala Sekolah	7
V. Spesifikasi Sistem Komputer yang digunakan.....	8
VI. Perangkat Lunak yang digunakan.....	8
VII. Prosedur Penggunaan Bagi Administrator	8
VIII. Prosedur Penggunaan Bagi Tata Usaha.....	13
IX. Prosedur Penggunaan Bagi Kepala Sekolah	17
X. Pengujian Unit <i>Login</i>	24
XI. Pengujian Unit <i>Form</i> Tata Usaha.....	25
XII. Pengujian Unit <i>Form</i> Siswa.....	26
XIII. Pengujian Unit <i>Form</i> Jurusan.....	26
XIV. Pengujian Unit <i>Form</i> Kelas	28
XV. Pengujian Unit <i>Form</i> Tahun Ajaran.....	28
XVI. Pengujian Unit Pembayaran	29
XVII. <i>Source Code</i>	30

I. Pendahuluan

SMK Jakarta Timur 2 merupakan lembaga pendidikan swasta yang terletak di Jakarta Timur. Berdasarkan peninjauan lokasi secara langsung di SMK Jakarta Timur 2 masih menggunakan sistem manual. Salah satu sistem yang masih manual yaitu pada pembayaran SPP. Dimana pada proses pengelolaan data pembayaran SPP masih sederhana. Walaupun sudah didukung dengan komputer tapi hanya menggunakan *Microsoft Excel* sebagai media pembuatan laporan pembayaran SPP dan sering terjadi kesalahan dalam pengelolaan data pembayaran SPP.

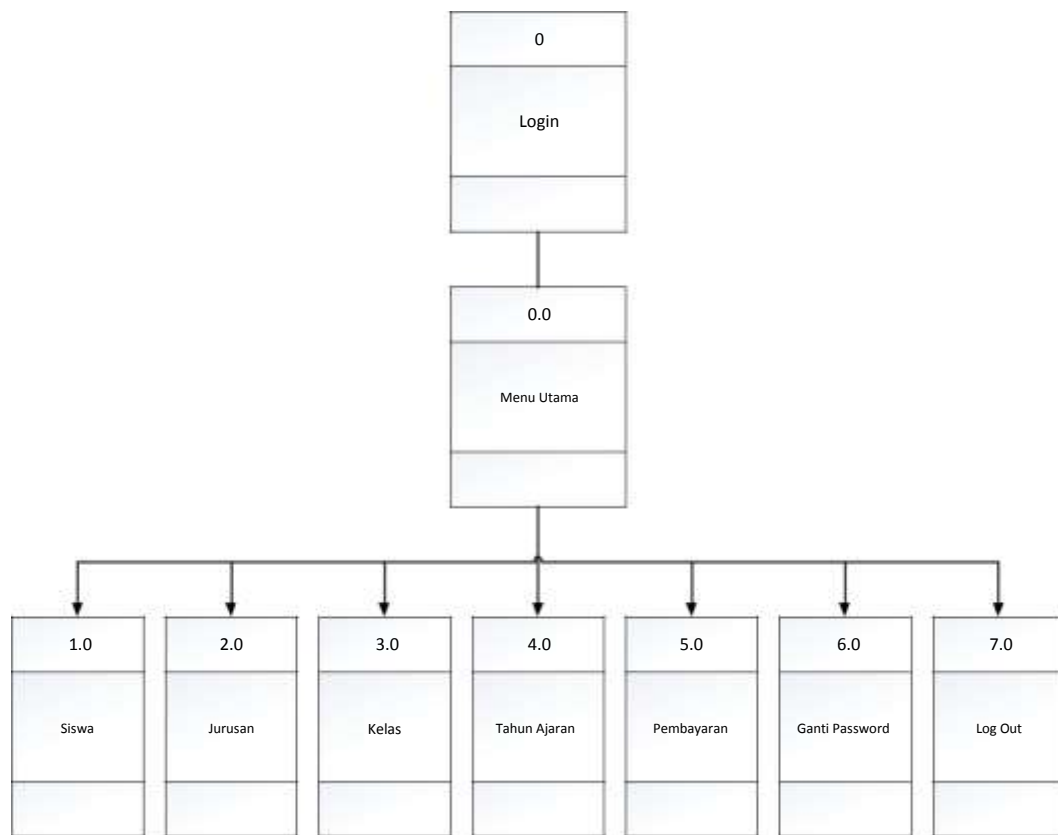
Dengan adanya permasalahan yang seringkali terjadi maka dibutuhkan suatu program pembayaran SPP yang terkomputerisasi, sehingga dapat meminimalisir kesalahan-kesalahan yang terjadi, dan dapat meringankan bagian tata usaha dalam mengelola data pembayaran serta menghasilkan laporan yang cepat dan akurat.

II. HIPO Administrator



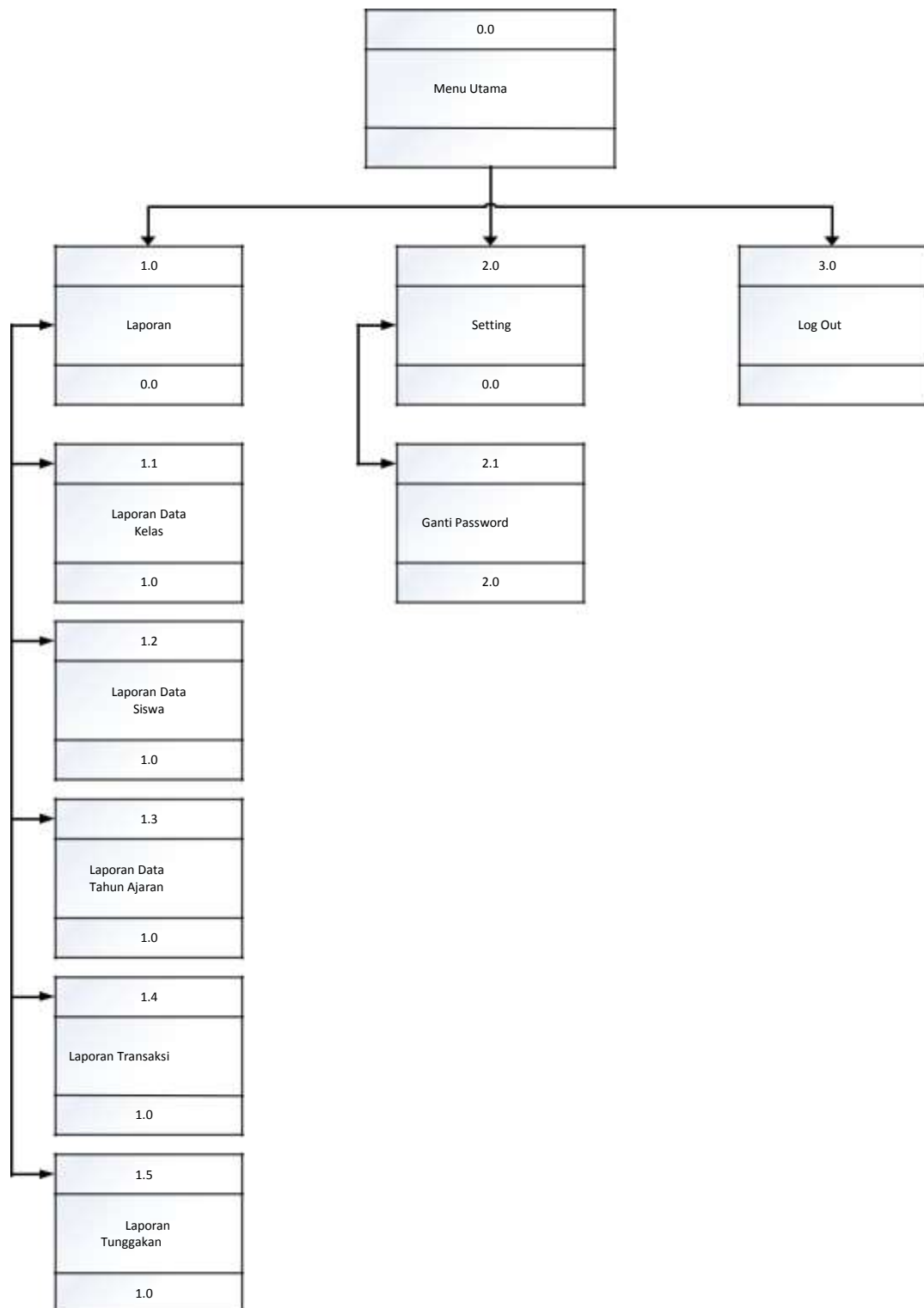
Gambar 1. HIPO Administrator

III. HIPO Tata Usaha



Gambar 2. HIPO Tata Usaha

IV. HIPO Kepala Sekolah



Gambar 3. HIPO Kepala Sekolah

V. Spesifikasi Sistem Komputer yang digunakan

Perangkat keras (*hardware*) yang digunakan untuk mendukung program pembayaran SPP ini yaitu:

1. *Processor : Intel Pentium Inside*
2. *RAM : 6GB*
3. *Harddisk : 80 GB*
4. *Monitor: 10"*
5. *Keyboard : Standar Keyboard*
6. *Mouse: Standar Mouse*
7. *Printer: Epson L360*

VI. Perangkat Lunak yang digunakan

Perangkat lunak (*software*) yang digunakan untuk mendukung program pembayaran SPP ini yaitu:

1. Sistem Operasi : *Microsoft Windows 2008*
2. Bahasa Pemrograman : *NetBeans IDE 8.2*
3. Basis Data : *PhpMyadmin 127.0.0.1*

VII. Prosedur Pengguna Bagi Administrator

1. Administrator menjalankan program pembayaran SPP yang terhubung dengan *XAMPP*. Lalu muncul seperti gambar di bawah ini. Administrator diminta untuk memasukkan *username* dan *password* untuk dapat masuk ke program pembayaran SPP.

Gambar 4. Halaman *Login* Administrator

Setelah administrator berhasil *login*, administrator dapat menjalankan menu-menu yang terdapat di halaman menu utama administrator.

2. Ketika administrator berhasil melakukan *login* maka administrator akan tampil halaman menu utama yang menampilkan pesan selamat datang dengan logo SMK Jakarta Timur 2 dan alamat SMK Jakarta Timur 2. Halaman ini juga berisi menu tata usaha, siswa, jurusan, kelas, tahun ajaran dan ganti *password*.



Gambar 5. Halaman Menu Utama Administrator

- Di menu tata usaha terdapat *form* tata usaha. Dimana administrator dapat mengelola data tata usaha.

Data Tata Usaha

NIP: 1908004
 Nama: budi
 Alamat: jakarta
 No. telp: 087654231456
 Jenis Kelamin: Laki-Laki
 Username: budi
 Password: ...

+ Batal Simpan Edit Hapus Keluar

Cari

NIP	Nama	Alamat	Jenis Kelamin	Telephone
1907001	arum	jakarta	Perempuan	083234321234
1907002	jaka	jakarta	Laki-Laki	084534523456
1908003	amel	bekasi	Perempuan	089678543142

Gambar 6. Halaman Menu Tata Usaha

- Di menu siswa terdapat *form* siswa. Dimana administrator dapat mengelola data siswa.

Data Siswa

NIS: 190003
 Tahun Ajaran: 2019 - 2020
 Nama Siswa: bayu
 Kelas: AK10a
 Jenis Kelamin: Laki-Laki
 Tanggal Lahir: 06 Mei 04
 Agama: Islam
 Alamat: bekasi

+ Batal Simpan Edit Hapus Keluar

Cari

NIS	Nama	Kelas	Jenis Kelamin	Tanggal Lahir	Agama	Alamat	Tahun Ajaran
190001	anisa	AK10a	Perempuan	2004-08-01	Islam	Jakarta	2019 - 2020
190002	andi	AK11a	Laki-Laki	2003-08-16	Islam	Jakarta	2019 - 2020

Gambar 7. Halaman Menu Siswa

5. Di menu jurusan terdapat *form* jurusan. Dimana administrator dapat mengelola data jurusan.

The screenshot shows a web application window titled "Data Jurusan". It features a form with two input fields: "Kode Jurusan" containing "PM" and "Nama Jurusan" containing "Pemasaran". Below the form are five buttons: "+ Batal", "Simpan", "Edit", "Hapus", and "Keluar". A search bar labeled "Cari" is positioned above a table. The table has two columns: "Kode Jurusan" and "Nama Jurusan". It contains two rows of data: one with "AK" and "Akutansi", and another with "AP" and "Administrasi Perkantoran".

Kode Jurusan	Nama Jurusan
AK	Akutansi
AP	Administrasi Perkantoran

Gambar 8. Halaman Menu Jurusan

6. Di menu kelas terdapat *form* kelas. Dimana administrator dapat mengelola data kelas.

The screenshot shows a web application window titled "Data Kelas". It features a form with three input fields: "Kode Kelas" containing "AK12a", "Nama Kelas" containing "12a", and "Nama Jurusan" with a dropdown menu showing "Akutansi". To the right of the "Kode Kelas" field is a link that says "Kode terisi otomatis". Below the form are four buttons: "+ Batal", "Simpan", "Hapus", and "Keluar". A search bar labeled "Cari" is positioned above a table. The table has three columns: "Kode Kelas", "Kode Jurusan", and "Nama Kelas". It contains two rows of data: one with "AK10a", "AK", and "10a", and another with "AK11a", "AK", and "11a".

Kode Kelas	Kode Jurusan	Nama Kelas
AK10a	AK	10a
AK11a	AK	11a

Gambar 9. Halaman Menu Kelas

7. Di menu tahun ajaran terdapat *form* tahun ajaran. Dimana administrator dapat mengelola data tahun ajaran.

Kode Tahun A...	Tahun Ajaran	Mulai Tahun A...	Akhir Tahun A...	Biaya SPP
1920	2019 - 2020	2019-07-19	2020-07-23	200000

Gambar 10. Halaman Menu Tahun Ajaran

8. Di menu ganti *password* terdapat *form* ganti *password*. Dimana administrator dapat melakukan ganti *password*.

ENTER

Perbarui Batal Tutup

Gambar 11. Halaman Menu Ganti *Password*

VIII. Prosedur Pengguna Bagi Tata Usaha

1. Tata usaha menjalankan program pembayaran SPP yang terhubung dengan *XAMPP*. Lalu muncul seperti gambar di bawah ini. Tata usaha diminta untuk memasukkan *username* dan *password* untuk dapat masuk ke program pembayaran SPP.



Gambar 12. Halaman *Login* Tata Usaha

Setelah tata usaha berhasil *login*, tata usaha dapat menjalankan menu-menu yang terdapat di halaman menu utama tata usaha.

2. Ketika tata usaha berhasil melakukan *login* maka tata usaha maka akan tampil halaman menu utama yang menampilkan pesan selamat datang dengan logo SMK Jakarta Timur 2 dan alamat SMK Jakarta Timur 2. Halaman ini juga berisi menu siswa, jurusan, kelas, tahun ajaran, pembayaran dan ganti *password*.



Gambar 13. Halaman Menu Utama Tata Usaha

3. Di menu siswa terdapat *form* siswa. Dimana tata usaha dapat mengelola data siswa.

The screenshot shows a web application window titled "Data Siswa". It contains a form for adding or editing student data. The form fields are: NIS (190003), Tahun Ajaran (2019 - 2020), Nama Siswa (bayu), Kelas (AK10a), Jenis Kelamin (Laki-Laki), Tanggal Lahir (06 Mei 04), Agama (Islam), and Alamat (bekasi). There are buttons for "Batal", "Simpan", "Edit", "Hapus", and "Keluar". A search bar labeled "Cari" is also present. Below the form is a table with the following data:

NIS	Nama	Kelas	Jenis Kelamin	Tanggal Lahir	Agama	Alamat	Tahun Ajaran
190001	anisa	AK10a	Perempuan	2004-08-01	Islam	jakarta	2019 - 2020
190002	andi	AK11a	Laki-Laki	2003-08-16	Islam	jakarta	2019 - 2020

Gambar 14. Halaman Menu Siswa

4. Di menu jurusan terdapat *form* jurusan. Dimana tata usaha dapat mengelola data jurusan.

The screenshot shows a web application window titled "Data Jurusan". It contains a form for adding or editing department data. The form fields are: Kode Jurusan (PM) and Nama Jurusan (Pemasaran). There are buttons for "Batal", "Simpan", "Edit", "Hapus", and "Keluar". A search bar labeled "Cari" is also present. Below the form is a table with the following data:

Kode Jurusan	Nama Jurusan
AK	Akutansi
AP	Administrasi Perkantoran

Gambar 15. Halaman Menu Jurusan

- Di menu kelas terdapat *form* kelas. Dimana tata usaha dapat mengelola data kelas.

Data Kelas

Kode Kelas: Kode terisi otomatis

Nama Kelas:

Nama Jurusan:

Cari:

Kode Kelas	Kode Jurusan	Nama Kelas
AK10a	AK	10a
AK11a	AK	11a

Gambar 16. Halaman Menu Kelas

- Di menu tahun ajaran terdapat *form* tahun ajaran. Dimana tata usaha dapat mengelola data tahun ajaran.

Data Tahun Ajaran

Kode Tahun Ajaran:

Tahun Ajaran: Tahun Ajaran Terisi Berdasarkan Tahun Sekarang

Mulai Tahun Ajaran:

Akhir Tahun Ajaran:

Biaya SPP (Bulan):

Cari:

Kode Tahun A...	Tahun Ajaran	Mulai Tahun A...	Akhir Tahun A...	Biaya SPP
1920	2019 - 2020	2019-07-19	2020-07-23	200000

Gambar 17. Halaman Menu Tahun Ajaran

7. Di menu pembayaran terdapat *form* pembayaran. Dimana tata usaha dapat melakukan transaksi pembayaran SPP.



The screenshot shows a web application window titled "Transaksi Pembayaran SPP". It contains several input fields for transaction details: "Tanggal" (Date) set to 26 Ago 19, "Nama" (Name) set to arda, "Bulan" (Month) set to Juli, "Jd Bayar" (Payment ID) set to 200188888, "Kelas" (Class) set to AK334, "Baya" (Amount) set to 200000, "NIS" (Student ID) set to 200001, and "TA" (Year) set to 2019 - 2020. There are buttons for "Cari" (Search) and "Simpan" (Save). Below the form is a table with columns "Bulan" and "Bayar", showing "September" and "200000". At the bottom right, a blue box displays the payment summary: "Jumlah Bayar" (Total Payment) as Rp. 200000, "Uang Bayar" (Payment) as Rp. 200000, and "Uang Kembali" (Change) as Rp. 0.

Gambar 18. Halaman Menu Pembayaran

8. Di menu ganti *password* terdapat *form* ganti *password*. Dimana tata usaha dapat melakukan ganti *password*.



The screenshot shows a web application window titled "Ganti Password". It contains four input fields: "NIP" (1907001), "Password Lama" (Old Password), "Password Baru" (New Password), and "Konfirm Password" (Confirm Password). There is an "ENTER" button next to the "Password Lama" field. At the bottom, there are three buttons: "Perbarui" (Update), "Batal" (Cancel), and "Tutup" (Close).

Gambar 19. Halaman Menu Ganti *Password*

IX. Prosedur Pengguna Bagi Kepala Sekolah

1. Kepala sekolah menjalankan program pembayaran SPP yang terhubung dengan *XAMPP*. Lalu muncul seperti gambar di bawah ini. Kepala sekolah diminta untuk memasukkan *username* dan *password* untuk dapat masuk ke program pembayaran SPP.

The image shows a web browser window displaying a login form titled "FORM LOGIN PEMBAYARAN SPP" in a green header. Below the header is the logo of SMK Jakarta Timur 2, which features a shield with a book, a mountain, and the school's name. Under the logo are two input fields labeled "USERNAME" and "PASSWORD". At the bottom of the form is a "Login" button.

Gambar 20. Halaman *Login* Kepala Sekolah

Setelah kepala sekolah berhasil *login*, tata usaha dapat menjalankan menu-menu yang terdapat di halaman menu utama kepala sekolah.

2. Ketika kepala sekolah berhasil melakukan *login* maka kepala sekolah maka akan tampil halaman menu utama yang menampilkan pesan selamat datang dengan logo SMK Jakarta Timur 2 dan alamat SMK Jakarta Timur 2. Halaman ini juga berisi menu laporan dan *setting*.



Gambar 21. Halaman Menu Utama Kepala Sekolah

3. Di menu laporan terdapat sub menu laporan-laporan. Dimana kepala sekolah dapat memilih laporan apa saja yang akan di cetak. Laporan- Laporan yang dapat dicetak yaitu; laporan data kelas, laporan data siswa, laporan data tahun ajaran, laporan transaksi dan laporan tunggakan.
- a. Laporan Data Kelas



SMK Jakarta Timur 2

LAPORAN DATA KELAS

No.	Kode Kelas	Nama Kelas
1	AK10a	10a
2	AK11a	11a
3	AK12a	12a
4	AP10a	10a
5	AP10b	10b

Jakarta, 21 September 2019
Kepala Sekolah

(.....)

Gambar 22. Laporan Data Kelas

- b. Laporan Data Siswa

Pada laporan data siswa kepala sekolah dapat mencetak berdasarkan per kelas, per nis, per tahun ajaran dan juga per jenis kelamin.

1. Laporan Data Siswa Per Kelas



SMK Jakarta Timur 2

LAPORAN DATA SISWA AK10a

No.	NIS	Nama	Kelas	Jenis Kelamin	Tanggal Lahir	Agama	Alamat	Tahun Ajaran
1	190002	andi	AK10a	Laki-Laki	30/09/2003	Islam	jakarta	2019 - 2020
2	190001	anisa	AK10a	Perempuan	01/09/2004	Islam	jakarta	2019 - 2020

Jakarta, 21 September 2019
Kepala Sekolah

(.....)

Gambar 23. Laporan Siswa Per Kelas

2. Laporan Data Siswa Per NIS



SMK Jakarta Timur 2
DATA SISWA

NIS : 190001
Nama : anisa
Alamat : jakarta
Tanggal Lahir : 01/08/2004
Kelas : AK10a

Jakarta, 21 September 2019
Kepala Sekolah

Gambar 24. Laporan Siswa Per
NIS 3. Laporan Data Siswa Per Tahun Ajaran



SMK Jakarta Timur 2

LAPORAN DATA SISWA PER TAHUN AJARAN 2019 - 2020

No.	NIS	Nama	Kelas	Jenis Kelamin	Tanggal Lahir	Agama	Alamat	Tahun Ajaran
1	190003	caca	AK11a	Perempuan	03/08/2003	Islam	jakarta	2019 - 2020
2	190002	andi	AK10a	Laki-Laki	30/08/2003	Islam	jakarta	2019 - 2020
3	190001	anisa	AK10a	Perempuan	01/08/2004	Islam	jakarta	2019 - 2020

Jakarta, 21 September 2019
Kepala Sekolah

Gambar 25. Laporan Siswa Per Tahun Ajaran

4. Laporan Data Siswa Per Jenis Kelamin



SMK Jakarta Timur 2

LAPORAN DATA SISWA Laki-Laki

No.	NIS	Nama	Kelas	Jenis Kelamin	Tanggal Lahir	Agama	Alamat	Tahun Ajaran
1	190002	andi	AK10e	Laki-Laki	30/08/2003	Islam	jakarta	2019 - 2020

Jakarta, 21 September 2019
Kepala Sekolah

(.....)

Gambar 26. Laporan Siswa Per Jenis Kelamin

c. Laporan Data Tahun Ajaran



SMK Jakarta Timur 2

LAPORAN DATA TAHUN AJARAN

No.	Kode	Tahun Ajaran	Biaya SPP
1	1920	2019 - 2020	200000

Jakarta, 21 September 2019
Kepala Sekolah

(.....)

Gambar 27. Laporan Data Tahun Ajaran

d. Laporan Transaksi

Pada laporan transaksi kepala sekolah dapat mencetak berdasarkan per kelas, per nis, per tahun ajaran dan juga per periode.

1. Laporan Transaksi Per Kelas



SMK Jakarta Timur 2

LAPORAN TRANSAKSI PER KELAS

Kelas : AK10a
Tahun Ajaran : 2019 - 2020

No.	No Pembayaran	Tanggal	NIS	Nama	Jumlah Bulan	Jumlah Bayar	NIP
1	1908160001	16/08/2019	190001	arisa	1	Rp 200.000,00	1907001
2	1908160002	16/08/2019	190001	arisa	1	Rp 200.000,00	1907001
3	1908160003	16/08/2019	190001	arisa	1	Rp 200.000,00	1907001
4	1908200005	20/08/2019	190001	arisa	1	Rp 200.000,00	1907001
5	1908200004	20/08/2019	190002	andi	1	Rp 200.000,00	1907001
6	1908220007	22/08/2019	190002	andi	1	Rp 200.000,00	1907002
7	1908220008	22/08/2019	190002	andi	2	Rp 200.000,00	1907001
8	1908220009	22/08/2019	190002	andi	2	Rp 200.000,00	1907001

Jakarta, 21 September 2019
Kepala Sekolah

Gambar 28. Laporan Transaksi Per Kelas 2. Laporan Transaksi Per Nis



SMK Jakarta Timur 2

LAPORAN TRANSAKSI PER NIS

NIS : 190001
Nama : arisa
Kelas : AK10a

No.	No Pembayaran	Tanggal	Tahun Ajaran	NIP	Nama Bulan	Jumlah Bayar
1	1908160001	16/08/2019	2019 - 2020	1907001	Juli	Rp 200.000,00
2	1908160002	16/08/2019	2019 - 2020	1907001	Agustus	Rp 200.000,00
3	1908160003	16/08/2019	2019 - 2020	1907001	September	Rp 200.000,00
4	1908200005	20/08/2019	2019 - 2020	1907001	Oktober	Rp 200.000,00
Jumlah Bulan					4	
Jumlah Bayar						Rp 800.000,00

Jakarta, 21 September 2019
Kepala Sekolah

Gambar 29. Laporan Transaksi Per NIS

3. Laporan Transaksi Per Tahun Ajaran



SMK Jakarta Timur 2

LAPORAN TRANSAKSI PER TAHUN AJARAN 2019 - 2020

No.	No Pembayaran	Tanggal	NIS	Jumlah Bulan	Jumlah Bayar	NIP
1	1908160001	16/08/2019	190001	1	Rp 200.000,00	1907001
2	1908160002	16/08/2019	190001	1	Rp 200.000,00	1907001
3	1908160003	16/08/2019	190001	1	Rp 200.000,00	1907001
4	1908200005	20/08/2019	190001	1	Rp 200.000,00	1907001
5	1908200004	20/08/2019	190002	1	Rp 200.000,00	1907001
6	1908220007	22/08/2019	190002	1	Rp 200.000,00	1907002
7	1908220008	22/08/2019	190002	2	Rp 200.000,00	1907001
8	1908220009	22/08/2019	190002	2	Rp 200.000,00	1907001
9	1908210006	21/08/2019	190003	1	Rp 200.000,00	1907001
10	1909210010	21/09/2019	190003	1	Rp 200.000,00	1907001
11	1909210011	21/09/2019	190003	1	Rp 200.000,00	1907001
TOTAL					Rp 2.200.000,00	

Jakarta, 21 September 2019
Kepala Sekolah

(.....)

Gambar 30. Laporan Transaksi Per Tahun Ajaran

4. Laporan Transaksi Per Periode



SMK Jakarta Timur 2

LAPORAN TRANSAKSI PER PERIODE 2019-08-01 s/d 2019-09-01

No.	No Pembayaran	Tanggal	NIS	Jumlah Bulan	Jumlah Bayar	NIP
1	1908160001	16/08/2019	190001	1	Rp 200.000,00	1907001
2	1908160002	16/08/2019	190001	1	Rp 200.000,00	1907001
3	1908160003	16/08/2019	190001	1	Rp 200.000,00	1907001
4	1908200005	20/08/2019	190001	1	Rp 200.000,00	1907001
5	1908200004	20/08/2019	190002	1	Rp 200.000,00	1907001
6	1908220007	22/08/2019	190002	1	Rp 200.000,00	1907002
7	1908220008	22/08/2019	190002	2	Rp 200.000,00	1907001
8	1908220009	22/08/2019	190002	2	Rp 200.000,00	1907001
9	1908210006	21/08/2019	190003	1	Rp 200.000,00	1907001
TOTAL					Rp 1.800.000,00	

Jakarta, 21 September 2019
Kepala Sekolah

(.....)

Gambar 31. Laporan Transaksi Per Periode

e. Laporan Tunggakan



SMK Jakarta Timur 2
LAPORAN TUNGGAKAN

NIS	Nama Siswa	Kelas	Tahun Ajaran	Jumlah Nunggak (Bulan)	Biaya SPP
190003	caca	AK11a	2019 - 2020	1	200000

Jakarta, 21 September 2019
Kepala Sekolah

{.....}

Gambar 32. Laporan Tunggakan

4. Di menu *setting* terdapat sub menu ganti *password*, jika kepala sekolah ingin melakukan klik ganti *password* maka kepala sekolah harus mengklik menu ganti password, setelah itu akan tampil *form* ganti *password*.



Gambar 33. Ganti Password

X. Pengujian Unit Login

No.	Skenario pengujian	Test case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1.	<i>Username</i> dan <i>Password</i> tidak diisi kemudian klik <i>login</i> .	<i>Username</i> : (kosong) <i>Password</i> : (kosong)	Sistem menolak akses <i>user</i> dan menampilkan “Anda belum memasukan <i>Username!</i> ”.	Sesuai harapan	Valid
2.	<i>Username</i> diisi dan <i>Password</i> tidak diisi kemudian klik <i>login</i> .	<i>Username</i> : (diisi) <i>Password</i> : (kosong)	Sistem menolak akses <i>user</i> dan menampilkan “Anda belum memasukan <i>Password!</i> ”.	Sesuai harapan	Valid
3.	<i>Username</i> tidak diisi dan <i>Password</i> diisi kemudian klik <i>login</i> .	<i>Username</i> : (kosong) <i>Password</i> : (diisi)	Sistem menolak akses <i>user</i> dan menampilkan “Anda belum memasukan <i>Username!</i> ”.	Sesuai harapan	Valid
4.	<i>Username</i> salah dan <i>Password</i> benar kemudian klik <i>login</i> .	<i>Username</i> : arumm (salah) <i>Password</i> : 123 (benar)	Sistem menolak akses <i>user</i> dan menampilkan “ <i>Username Salah</i> ”.	Sesuai harapan	Valid
5.	<i>Username</i> benar dan <i>Password</i> salah kemudian klik <i>login</i> .	<i>Username</i> : arum (benar) <i>Password</i> : 12345 (salah)	Sistem menolak akses <i>user</i> dan menampilkan “ <i>Username</i> atau <i>Password</i> salah”.	Sesuai harapan	Valid
6.	<i>Username</i> salah dan <i>Password</i> salah kemudian klik <i>login</i> .	<i>Username</i> : arumm (salah) <i>Password</i> : 12345 (salah)	Sistem menolak akses <i>user</i> dan menampilkan “ <i>Username Salah</i> ”.	Sesuai harapan	Valid

7.	<i>Username</i> benardan <i>Password</i> benar kemudian klik <i>login</i> .	<i>Username</i> : arum (benar) <i>Password</i> : 123 (benar)	Sistem menerima akses <i>login</i> , menampilkan pesan “Selamat datang di aplikasi pembayaran spp admin” lalu langsung menampilkan menu utama.	Sesuai harapan	Valid
----	--	---	--	----------------	-------

XI. Pengujian Unit *Form* Tata Usaha

No.	Skenario pengujian	Test case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1.	Klik tambah, kemudian semua <i>field</i> dikosongkan lalu klik simpan.	Semua <i>field</i> kosong	Sistem menolak penambahan data dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid
2.	Klik tambah, kemudian kosongkan salah satu <i>field</i> dan isi <i>field</i> lainnya lalu klik simpan.	<i>Field</i> Nama, Alamat, No. Telp, Jenis Kelamin, <i>Username</i> (diisi) <i>Password</i> : (kosong)	Sistem menolak penambahan data dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid
3.	Klik tambah, kemudian isi semua <i>field</i> lalu klik simpan.	Semua <i>field</i> diisi	Sistem menerima dan menyimpan data tata usaha dan menampilkan “Simpan Data Berhasil”.	Sesuai harapan	Valid

XII. Pengujian Unit *Form* Siswa

No.	Skenario pengujian	Test case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1.	Klik tambah, kemudian semua <i>field</i> dikosongkan lalu klik simpan.	Semua <i>field</i> kosong	Sistem menolak penambahan data dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid
2.	Klik tambah, kemudian kosongkan salah satu <i>field</i> dan isi <i>field</i> lainnya lalu klik simpan.	<i>Field</i> Nama Siswa, Kelas, Jenis Kelamin, Tanggal Lahir, Agama (diisi) Alamat : (kosong)	Sistem menolak penambahan data dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid
3.	Klik tambah, kemudian isi semua <i>field</i> lalu klik simpan.	Semua <i>field</i> diisi	Sistem menerima dan menyimpan data siswa dan menampilkan “Simpan Data Berhasil”.	Sesuai harapan	Valid

XIII. Pengujian Unit *Form* Jurusan

No.	Skenario pengujian	Test case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1.	Klik tambah, kemudian semua <i>field</i> dikosongkan lalu klik simpan.	Semua <i>field</i> kosong	Sistem menolak penambahan data dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid

2.	Klik tambah, kemudian kosongkan <i>field</i> Nama Jurusan lalu klik simpan.	Kode Jurusan: (diisi) Nama Jurusan : (kosong)	Sistem menolak penambahan data dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid
3.	Klik tambah, kemudian isi semua <i>field</i> lalu klik simpan.	Kode Jurusan dan Nama Jurusan (diisi)	Sistem menerima dan menyimpan data siswa dan menampilkan “Simpan Data Berhasil”.	Sesuai harapan	Valid

XIV. Pengujian Unit *Form* Kelas

No.	Skenario pengujian	Test case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1.	Klik tambah, kemudian kosongkan semua <i>field</i> lalu klik simpan.	Semua <i>field</i> kosong	Sistem menolak penambahan data dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid
2.	Klik tambah, kemudian kosongkan salah satu <i>field</i> dan isi <i>field</i> lainnya lalu klik simpan.	<i>Field</i> Nama Kelas (diisi) Nama Jurusan : (tidak dipilih)	Sistem menolak penambahan data dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid
3.	Klik tambah, kemudian isi semua <i>field</i> lalu klik simpan.	Semua <i>field</i> diisi	Sistem menerima dan menyimpan data siswa dan menampilkan “Simpan Data Berhasil”.	Sesuai harapan	Valid

XV. Pengujian Unit *Form Tahun Ajaran*

No.	Skenario pengujian	Test case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1.	Klik tambah, kemudian semua <i>field</i> dikosongkan lalu klik simpan.	Semua <i>field</i> kosong	Sistem menolak penambahan data dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid
2.	Klik tambah, kemudian kosongkan salah satu <i>field</i> dan isi <i>field</i> lainnya lalu klik simpan.	<i>Field</i> Kode Tahun Ajaran, Tahun Ajaran, Mulai Tahun Ajaran, Akhir Tahun Ajaran (diisi) Biaya SPP (bulan) : (kosong)	Sistem menolak penambahan data dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid
3.	Klik tambah, kemudian isi semua <i>field</i> lalu klik simpan.	Semua <i>field</i> diisi	Sistem menerima dan menyimpan data siswa dan menampilkan “Simpan Data Berhasil”.	Sesuai harapan	Valid

XVI. Pengujian Unit Pembayaran

No.	Skenario pengujian	Test case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1.	Klik baru, kemudian semua <i>field</i> dikosongkan lalu klik tambah.	Semua <i>field</i> kosong	Sistem menolak penambahan data pembayaran dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid
2.	Klik baru, kemudian kosongkan salah satu <i>field</i> dan isi <i>field</i> lainnya lalu klik tambah.	<i>Field</i> Tanggal, Id Bayar, NIS, Nama, Kelas, TA, Biaya (diisi) Bulan : (kosong)	Sistem menolak penambahan data dan menampilkan “Silahkan Lengkapi Data!!”.	Sesuai harapan	Valid
3.	Klik baru, kemudian isi semua <i>field</i> tetapi uang bayar kosong, lalu klik cetak.	<i>Field</i> Tanggal, Id Bayar, NIS, Nama, Kelas, TA, Biaya, Bulan (diisi) Uang Bayar (kosong)	Sistem menolak dan menampilkan “Uang Bayar Kurang!!”.	Sesuai harapan	Valid
4.	Klik baru, kemudian isi semua <i>field</i> , uang bayar diisi, lalu klik cetak.	Semua <i>field</i> diisi	Sistem menerima dan menyimpan data pembayaran dan menampilkan “Simpan Data Berhasil”.	Sesuai harapan	Valid

XVII. Source Code

1. Source Code Login

```
package view;

import entitas.Entitas_TU; import
java.awt.event.KeyEvent; import
java.util.ArrayList; import
java.util.List;
import javax.swing.JFrame;
import javax.swing.JOptionPane;
import method.User_Method;

import java.sql.*;
import javax.swing.ImageIcon;
import javax.swing.JFrame;
import javax.swing.JOptionPane;

/**
 *
 * @author
 * Asus */
public class form_login extends javax.swing.JFrame {

    List<Entitas_TU> listLogin = new ArrayList<>();
    User_Method uc = new User_Method();
    Entitas_TU eu = new Entitas_TU();
    kon kon = new kon();

    public form_login() {
        initComponents();
        ImageIcon ico = new
        ImageIcon("src/img/logo_1.png");
        setIconImage(ico.getImage()); kon.setKoneksi();
        setLocationRelativeTo(this);
    }

    void Bersih() {
        tkodeuser.setText("");
        tpassword.setText("");
        tlevel.setText("");
    }
}
```

```

void bersih() {
    tkodeuser.setText("");
    tpassword.setText("");
    tlevel.setText("");
}

private void prosesLogin() {
    String kodeuser = "";
    String password = "";
    kodeuser = tkodeuser.getText();
    try {
        String sql = "Select
nip,nama_tu,alamat,no_telp,j_kelamin,username,password,level from user
where username='" + kodeuser + "' AND password='" +
tpassword.getText() + "'";
        kon.rs = kon.st.executeQuery(sql);
        if (kon.rs.next()) {
            if (tlevel.getText().equals("kepsek")) {
                Dashboard_kepsek menu = new Dashboard_kepsek();
                String nama = kon.rs.getString("username");
                String nip = kon.rs.getString("nip");
                String hak_akses = kon.rs.getString("level");
                JOptionPane.showMessageDialog(this, "Selamat datang Di
Aplikasi Pembayaran SPP " + nama, "Welcome",
JOptionPane.INFORMATION_MESSAGE);
                //String id_user=kon.rs.getString("kd_user");
                tlevel.setText(hak_akses);
                tkodeuser.setText(nama);
                lblhak1.setText(hak_akses);
                idtf.setText(nip);

                menu.setLocationRelativeTo(null);
                menu.setVisible(true);
                menu.Nip = kon.rs.getString("nip");
                menu.Username = kon.rs.getString("username");
                menu.level = kon.rs.getString("level");

                this.dispose();
            } else if (tlevel.getText().equals("tata-usaha"))
            { Dashboard menu = new Dashboard();
                String nip = kon.rs.getString("nip");

```

```

        String nama = kon.rs.getString("username"); String pass =
        kon.rs.getString("password"); String hak_akses =
        kon.rs.getString("level");
        JOptionPane.showMessageDialog(this, "Selamat datang Di
Aplikasi Pembayaran SPP " + nama, "Welcome",
JOptionPane.INFORMATION_MESSAGE);
        tlevel.setText(hak_akses);
        tkodeuser.setText(nama);
        lblhak1.setText(hak_akses);
        idtf.setText(nip);

        menu.setLocationRelativeTo(null);
        menu.setVisible(true);
        menu.Nip = kon.rs.getString("nip");
        menu.Username = kon.rs.getString("username");
        menu.level = kon.rs.getString("level");

        this.dispose();
    } else if(tlevel.getText().equals("Administrator")){
        DashboardAdmin menu = new DashboardAdmin();
        String nip = kon.rs.getString("nip");
        String nama = kon.rs.getString("username");
        String hak_akses = kon.rs.getString("level");
        JOptionPane.showMessageDialog(this, "Selamat datang Di
Aplikasi Pembayaran SPP " + nama, "Welcome",
JOptionPane.INFORMATION_MESSAGE);
        tlevel.setText(hak_akses);
        tkodeuser.setText(nama);
        lblhak1.setText(hak_akses);
        idtf.setText(nip);

        menu.setLocationRelativeTo(null);
        menu.setVisible(true);
        menu.Nip = kon.rs.getString("nip");
        menu.Username = kon.rs.getString("username");
        menu.level = kon.rs.getString("level");

        this.dispose();
    }
} else if (tpassword.getText().isEmpty()) {
    JOptionPane.showMessageDialog(this,"Andabelum

```

```

memasukan Password !", "Warning",
JOptionPane.INFORMATION_MESSAGE);
    } else {
        JOptionPane.showMessageDialog(null, "Username atau
Password Salah");
        bersih();
        tkodeuser.setEnabled(true);
        tkodeuser.requestFocus();
    }

    } catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e);
    }
}
@SuppressWarnings("unchecked")
private void loginActionPerformed(java.awt.event.ActionEvent evt)
    { // TODO add your handling code here:

    String username = "";
    username = tkodeuser.getText();
    try {
        String sql = "Select *from user where username=" + username
+ """;
        kon.rs = kon.st.executeQuery(sql);

        if (kon.rs.next()) {
            tkodeuser.setEnabled(false);
            tlevel.setText(kon.rs.getString("level"));
            tpassword.setEnabled(true);
            tpassword.requestFocus();
            tkodeuser.setEnabled(false);
            prosesLogin();
        } else {
            if (tkodeuser.getText().isEmpty()) {
                JOptionPane.showMessageDialog(null, "Anda belum
memasukan Username !", "Warning",
JOptionPane.INFORMATION_MESSAGE);
            } else if (tkodeuser.getText().isEmpty() ||
tpassword.getText().isEmpty()) {
                JOptionPane.showMessageDialog(null, "Anda belum
memasukan Username dan Password !", "Warning",

```

```

OptionPane.INFORMATION_MESSAGE);
        } else {

            JOptionPane.showMessageDialog(null, "User name
Salah");

            Bersih();
            tkodeuser.setEnabled(true);
            tkodeuser.requestFocus();
        }
    }
} catch (SQLException e) {
    JOptionPane.showMessageDialog(null, e);
}

}

private void tpasswordKeyReleased(java.awt.event.KeyEvent evt)
{ // TODO add your handling code here:
}

private void tpasswordKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if (evt.getKeyCode() == 10) {
        loginActionPerformed(null);
    }
}

private void tkodeuserKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if (evt.getKeyCode() == KeyEvent.VK_ENTER)
    {
        tpassword.requestFocus();
    }
}

public static void main(String args[]) {
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Windows".equals(info.getName())) {

```

```

        javax.swing.UIManager.setLookAndFeel(info.getClassName());
        break;
    }
}
} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(form_login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(form_login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(form_login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(form_login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
}
//</editor-fold>
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new form_login().setVisible(true);
        }
    });
}
}

```

2. **Source Code Form**

Siswa package view;

```

import javax.swing.table.DefaultTableModel;
import entitas.*;
import java.sql.Connection; import
java.sql.ResultSet; import
java.sql.SQLException; import
java.sql.Statement; import
java.text.DateFormat; import
java.text.SimpleDateFormat;
import method.*;

```

```

import java.util.ArrayList;
import java.util.Date;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import method.koneksi;
import java.util.Locale;
import java.util.Calendar;
import java.util.Date;
import entitas.Bantuan;
import java.awt.event.KeyEvent;
import java.text.ParseException;

/**
 *
 * @author
 * Asus */
public class Master_Siswa extends javax.swing.JFrame {

    private DefaultTableModel model;
    List<Entitas_Siswa> listSiswa = new ArrayList<>();
    Entitas_Siswa es = new Entitas_Siswa();
    Siswa_Method sm = new Siswa_Method();
    Connection con = null;
    Statement st = null;
    ResultSet rs = null;
    String sql = null;
    List<Entitas_TA> listTA = new ArrayList<>();
    Entitas_TA et = new Entitas_TA();
    TA_Method tm = new TA_Method();
    kon kon = new kon();

    List<Entitas_Kelas> listKls = new ArrayList<>();
    Entitas_Kelas ek = new Entitas_Kelas();
    Kelas_Method km = new Kelas_Method();
    SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
    String kodeTA;

    public Master_Siswa() {
        initComponents();
    }

```

```

        try {
            st = koneksi.getKoneksi().createStatement();
        } catch (SQLException ex) {

Logger.getLogger(Pembayaran.class.getName()).log(Level.SEVERE, null,
ex);
        }
        BuatTabel();
        tampil();
        kon.setKoneksi();
        tahun();
        NamaJenis();
        edtanggal.setDate(new Date());
        Bersih();
        SiapIsi(false);
        tombolAwal();
        EdNis.setEnabled(false);
        edtanggal.setDate(null);
        tahun.setVisible(false);
        ttahun1.setVisible(false);
        txtambilno.setVisible(false);
        setLocationRelativeTo(this);

    }

    void NamaJenis() {
        try {
            String sql = "Select * From kelas";
            kon.rs = kon.st.executeQuery(sql);
            while (kon.rs.next()) {
                cbmkelas.addItem(kon.rs.getString("kd_kelas"));

            }
        } catch (SQLException e) {
            System.out.println("Koneksi Gagal" + e.toString());
        }
    }

    @SuppressWarnings("unchecked")
    void tambah(){

```

```

Locale lokal = null;
String pola = "yyyy";
Date tahunAwal = new Date();

int jmlTambahTahun = 1;
String satuan = "tahun";
Calendar cal = Bantuan.tambahTahun(tahunAwal,
jmlTambahTahun, satuan);
String tahunStr = Bantuan.tampilkanTahun(cal.getTime(), pola,
lokal);

cal = Bantuan.tambahTahun(tahunAwal, jmlTambahTahun,
satuan);
tahunStr = Bantuan.tampilkanTahun(cal.getTime(), pola, lokal);

ttahun1.setText((Bantuan1.tampilkanTahun(tahunAwal, pola,
lokal).substring(2, 4)) + (tahunStr).substring(0, 2));
ttahun.setText(Bantuan.tampilkanTahun(tahunAwal, pola, lokal) +
" - " + tahunStr);
}
private void btntambahActionPerformed(java.awt.event.ActionEvent
evt) {
// TODO add your handling code here:
if (btntambah.getText().equalsIgnoreCase("tambah"))
{ btntambah.setText("Batal");
Bersih();
SiapIsi(true);
ednama.requestFocus();
btntambah.setEnabled(true);
btnsimpan.setEnabled(true);
btnhapus.setEnabled(false);
btncari.setEnabled(false);
btnkeluar.setEnabled(false);
ttahun.setEnabled(false);
cari.setEnabled(false);
buatkode();
tambah();
} else {
btntambah.setText("Tambah");
Bersih();
SiapIsi(false);

```

```

        tombolAwal();
        cari.setEnabled(true);
    }
}

private void btnsimpanActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:
    if (EdNis.getText().isEmpty() || ednama.getText().isEmpty()
        || cbmkelas.getSelectedIndex() == 0 ||
cbmJK.getSelectedIndex() == 0
        || cbmAgama.getSelectedIndex() == 0 ||
edAlamat.getText().isEmpty()) {
        JOptionPane.showMessageDialog(null, "Silahkan Lengkapi
Data !!");
    } else { es.setNis(EdNis.getText());
        es.setNama_siswa(ednama.getText());
        es.setKelas(cbmkelas.getSelectedItem().toString());
        es.setJ_kelamin(cbmJK.getSelectedItem().toString());
        es.setTgl_lahir(sdf.format(edtanggal.getDate()));
        es.setAgama(cbmAgama.getSelectedItem().toString());
        es.setAlamat(edAlamat.getText());
        es.setTahun_ajaran(ttahun1.getText());

        if (btntambah.getText().equalsIgnoreCase("batal"))
        { if (sm.tambah(es) == 1) {
            entitas.nunggak n = new nunggak();
            method.Nunggak_Method mn = new Nunggak_Method();
            n.setNis(EdNis.getText());
            mn.tambahstatus(n);
            JOptionPane.showMessageDialog(null, "Simpan Data
Berhasil");
        } else {
            JOptionPane.showMessageDialog(null, "Simpan Data
Gagal !!");
        }
        tampil();
    }

    if (btncancel.getText().equalsIgnoreCase("batal")) {

```

```

        if (sm.edit(es) == 1) {
            JOptionPane.showMessageDialog(null, "Edit Data
Berhasil");
        } else {
            JOptionPane.showMessageDialog(null, "Edit Data Gagal
!!");
        }
        tampil();
    }

    Bersih();
    SiapIsi(false);
    btntambah.setText("Tambah");
    btncari.setText("Edit");
    tombolAwal();
    cari.setEnabled(true);
}
}

```

```

private void btncariActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:
    es.setText(EdNis.getText());

    if (btncari.getText().equalsIgnoreCase("edit")) {
        btncari.setText("Batal");
        SiapIsi(true);
        EdNis.setEnabled(false);
        ednama.requestFocus();
        btntambah.setEnabled(false);
        btntambah.setText("Tambah");
        btncari.setEnabled(true);
        btncari.setText("Edit");
        btncari.setEnabled(true);
        btncari.setEnabled(false);
        cari.setEnabled(false);
    } else {
        btncari.setText("Edit");
        Bersih();
        SiapIsi(false);
        tombolAwal();
    }
}

```

```

        cari.setEnabled(true);
    }
}

private void btnhapusActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:
    es.setNis(EdNis.getText());

    int pesan = JOptionPane.showConfirmDialog(null, "Apakah Anda
Yakin Ingin Menghapus Data Ini ??", "Konfirmasi",
JOptionPane.YES_NO_OPTION,
JOptionPane.QUESTION_MESSAGE);
    if (pesan == JOptionPane.YES_OPTION) {
        if (sm.hapus(es) == 1) {
            entitas.nunggak n = new nunggak();
            method.Nunggak_Method mn = new Nunggak_Method();
            n.setNis(EdNis.getText());
            mn.hapusstatus(n);
            JOptionPane.showMessageDialog(null, "Hapus Data
Berhasil");
            Bersih();
            SiapIsi(false);
            btntambah.setText("Tambah");
            tombolAwal();
            tampil();
            buatkode();
        } else {
            JOptionPane.showMessageDialog(null, "Hapus Data Gagal
!!");
        }
    }
}

private void btnkeluarActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:
    dispose();
}

private void tbldiswaliMouseClicked(java.awt.event.MouseEvent evt)

```

```

{
    // TODO add your handling code here: int baris =
    tblsiswa.getSelectedRow();
    EdNis.setText(tblsiswa.getModel().getValueAt(baris,
0).toString());
    ednama.setText(tblsiswa.getModel().getValueAt(baris,
1).toString());
    edAlamat.setText(tblsiswa.getModel().getValueAt(baris,
6).toString());
    ttahun.setText(tblsiswa.getModel().getValueAt(baris,
7).toString());
    cbmkelas.setSelectedItem(tblsiswa.getModel().getValueAt(baris,
2).toString());
    cbmJK.setSelectedItem(tblsiswa.getModel().getValueAt(baris,
3).toString());

    cbmAgama.setSelectedItem(tblsiswa.getModel().getValueAt(baris,
5).toString());
    String tgll = (String) tblsiswa.getValueAt(baris, 4);
    SimpleDateFormat date = new SimpleDateFormat("yyyy-MM-
dd");
    java.util.Date tanggal = null;
    try {
        tanggal = date.parse(tgll);
    } catch (ParseException ex) {

Logger.getLogger(Master_Siswa.class.getName()).log(Level.SEVERE,
null, ex);
    }
    edtanggal.setDate(tanggal);

    SiapIsi(false);
    btnsimpan.setEnabled(false);
    btntambah.setEnabled(true);
    btntambah.setText("Batal");
    btnhapus.setEnabled(true);
    btncancel.setEnabled(true);
}

private void cariKeyReleased(java.awt.event.KeyEvent evt) {

```

```

// TODO add your handling code here:
String find = null;
find = cari.getText();
model.getDataVector().removeAllElements();
model.fireTableDataChanged();
listSiswa.clear();
listSiswa = sm.cari(find);
for (int i = 0; i < listSiswa.size(); i++)
{
    Object[] a = new Object[8];
    a[0] = listSiswa.get(i).getNis();
    a[1] = listSiswa.get(i).getNama_siswa();
    a[2] = listSiswa.get(i).getKelas();
    a[3] = listSiswa.get(i).getJ_kelamin();
    a[4] = listSiswa.get(i).getTgl_lahir();
    a[5] = listSiswa.get(i).getAgama();
    a[6] = listSiswa.get(i).getAlamat();
    a[7] = listSiswa.get(i).getTahun_ajaran();
    model.addRow(a);
}
}

private void ttahunActionPerformed(java.awt.event.ActionEvent evt)
{
    // TODO add your handling code here:
}

private void cbmkelasActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here: if
    (cbmkelas.getSelectedIndex() == 0) {

    } else {
        tm.carikode(cbmkelas.getSelectedItem().toString());
    }
}

private void formWindowActivated(java.awt.event.WindowEvent
evt) {
    // TODO add your handling code here:
    this.setAlwaysOnTop(false);
}

```

```

    }

    private void ednamaKeyTyped(java.awt.event.KeyEvent evt) {
        // TODO add your handling code here:
        filterangka(evt);
    }

    private void
    cbmAgamaActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        edAlamat.requestFocus();
    }

    public static void main(String args[ ]) { /* Create and
        display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Master_Siswa().setVisible(true);
            }
        });
    }

    private void BuatTabel() {
        model = new DefaultTableModel();
        model.addColumn("NIS");
        model.addColumn("Nama");
        model.addColumn("Kelas");
        model.addColumn("Jenis Kelamin");
        model.addColumn("Tanggal Lahir");
        model.addColumn("Agama");
        model.addColumn("Alamat");
        model.addColumn("Tahun Ajaran");
        tblsiswa.setModel(model);
    }

    void filterangka(KeyEvent b){
        if(Character.isDigit(b.getKeyChar())){
            b.consume();
        }
    }

    private void tombolAwal() {

```

```

        btntambah.setEnabled(true);
        btnsimpan.setEnabled(false);
        btncancel.setEnabled(false);
        btnhapus.setEnabled(false);
        btnkeluar.setEnabled(true);
    }

```

```

private void tampil() {
    model.getDataVector().removeAllElements();
    model.fireTableDataChanged();
    listSiswa.clear();
    listSiswa = sm.tampil();
    for (int i = 0; i < listSiswa.size(); i++) {
        Object[] a = new Object[8];
        a[0] = listSiswa.get(i).getNis();
        a[1] = listSiswa.get(i).getNama_siswa();
        a[2] = listSiswa.get(i).getKelas();
        a[3] = listSiswa.get(i).getJ_kelamin();
        a[4] = listSiswa.get(i).getTgl_lahir();
        a[5] = listSiswa.get(i).getAgama();
        a[6] = listSiswa.get(i).getAlamat();
        a[7] = listSiswa.get(i).getTahun_ajaran();
        model.addRow(a);
    }
}

```

```

private void tahun(){
    Date now = new Date();
    SimpleDateFormat simpleDateFormat = new
SimpleDateFormat("yy");
    String thn = simpleDateFormat.format(now);
    tahun.setText(thn);
}

```

```

private void buatkode() {

    String thn = tahun.getText().substring(0, 2);
    String kode = (thn.substring(0, 2));
    try {
        String sql = "Select max(right(nis,4)) as no From siswa";
        rs = st.executeQuery(sql);
        while (rs.next()) {

```

```

if (rs.first() == false) {
    EdNis.setText(kode);
} else {
    rs.last();
    int auto_id = rs.getInt(1) + 1;
    String no = String.valueOf(auto_id);
    int noLong = no.length();
    for (int a = 0; a < 4 - noLong; a++) {
        no = "0" + no;
    }
    EdNis.setText(kode+ no);
    String baca="select*from siswa where nis In(Select
Max(nis)from siswa) order by nis desc";
    rs = st.executeQuery(baca);
    while(rs.next())
    {
        txtambilno.setText(rs.getString("nis"));
        String ambil = txtambilno.getText().substring(2,6);

        //--Penomoran ketika bernilai maksimal 999 maka akan
berulang ke 001
        int ambil2 = Integer.parseInt(ambil);
        if(ambil2 == 999){
            int auto = 000 + 1;
            String nomor = String.valueOf(auto);
            int noLong2 = nomor.length();
            for(int b=0;b<4-noLong2;b++){
                nomor= "0"+nomor;
            }
            EdNis.setText(kode+nomor);
            //--Penomoran ketika nilainya berisikan 1 sampai 998 dia
akan bertambah 1
        }else if(ambil2 > 0 && ambil2 < 999){
            int auto = ambil2+1;
            String nomor = String.valueOf(auto);
            int noLong2 = nomor.length();
            for(int b=0;b<4-noLong2;b++){
                nomor= "0"+nomor;
            }
            EdNis.setText(kode+nomor);
        }
    }
}

```

```

        }
    }
}
rs.close();
} catch (Exception e) {
    e.printStackTrace();
}
}

private void Bersih() {

    ednama.setText(null);
    EdNis.setText(null);
    cbmkelas.setSelectedIndex(0);
    ttahun.setText(null);
    cbmJK.setSelectedIndex(0);
    edtanggal.setDate(null);
    cbmAgama.setSelectedIndex(0);
    edAlamat.setText(null);

}

private void SiapIsi(boolean a) {

    ednama.setEnabled(a);
    cbmkelas.setEnabled(a);
    cbmJK.setEnabled(a);
    edtanggal.setEnabled(a);
    cbmAgama.setEnabled(a);
    edAlamat.setEnabled(a);
    ttahun.setEnabled(a);

}
}

```

3. *Source Code Form*

Pembayaran package view;

```

import entitas.*; import
java.io.File; import
java.sql.Connection; import
java.sql.ResultSet; import
java.sql.SQLException;

```

```

import java.sql.Statement;
import java.text.SimpleDateFormat;
import method.*;
import java.util.ArrayList;
import java.util.Date;
import java.util.HashMap;
import java.util.List;
import java.util.Locale;
import java.util.Map;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
import method.Siswa_Method;
import net.sf.jasperreports.engine.JasperCompileManager;
import net.sf.jasperreports.engine.JasperFillManager; import
net.sf.jasperreports.engine.JasperPrint;
import net.sf.jasperreports.engine.JasperReport; import
net.sf.jasperreports.engine.design.JasperDesign; import
net.sf.jasperreports.engine.xml.JRXmlLoader; import
net.sf.jasperreports.view.JasperViewer; import
method.koneksi;

/**
 *
 * @author
 * Asus */
public class Pembayaran extends javax.swing.JFrame {

    private DefaultTableModel model;
    String auto;
    Connection con = null;
    Statement st = null;
    ResultSet rs = null;
    String sql = null;
    String Bln;
    List<Entitas_Siswa> listSiswa = new ArrayList<>();
    Entitas_Siswa es = new Entitas_Siswa();
    Siswa_Method sm = new Siswa_Method();
    private DefaultTableModel modeltrnas;
    List<Entitas_Pembayaran> listBayar = new ArrayList<>();

```

```

Entitas_Pembayaran ep = new Entitas_Pembayaran();
SPP_Method sp = new SPP_Method();

List<Entitas_Trans> listSpp = new ArrayList<>();
Entitas_Trans et = new Entitas_Trans();
Trans_Method tm = new Trans_Method();

List<Entitas_TA> listTA = new ArrayList<>();
Entitas_TA eta = new Entitas_TA();
TA_Method tam = new TA_Method();
SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
JasperReport jasperReport1;
JasperDesign jasperDesign1;
JasperPrint jasperPrint1;
Map<String, Object> param = new HashMap<String,
Object>(); int total, jml;

public Pembayaran() {

    initComponents();
    Locale.setDefault(new Locale("in", "ID"));
    tgl.setDate(new Date());
    BuatTabel();
    tampil();
    tanggal();
    bulan();
    tahun();

    try {
        st = koneksi.getKoneksi().createStatement();
    } catch (SQLException ex) {

Logger.getLogger(Pembayaran.class.getName()).log(Level.SEVERE, null,
ex);
    }

    SiapIsi(false);
    btntambah.setEnabled(true);
    tanggal.setVisible(false);
    bln.setVisible(false);
    tahun.setVisible(false);

```

```

        txtambilno.setVisible(false);
        nip.setText(Dashboard.nip.getText());
    }
    @SuppressWarnings("unchecked")

    private void btntambahActionPerformed(java.awt.event.ActionEvent
    evt) {

        if (btntambah.getText().equalsIgnoreCase("Baru"))
        {
            btntambah.setText("Cetak");
            Bersih();
            buatkode();
            BuatTabelsiswa();
            tampilsiswa();
            SiapIsi(true);
            btntambah.setEnabled(false);
            btnhapus.setEnabled(false);
            totalbayar.setEnabled(false);
            kembali.setEnabled(false);
            btnsimpan.setEnabled(true);
            edcari.setEnabled(true);
            btnbatal.setEnabled(true);
            bulan.setEnabled(true);
            btnkeluar.setEnabled(false);
        } else {
            int ukem = Integer.parseInt(kembali.getText());
            int ubay = Integer.parseInt(bayar.getText());
            int total = Integer.parseInt(totalbayar.getText());
            int bayar1 = Integer.parseInt(jumbay.getText());
            int nol = 0;

            ep.setId_pembayaran(nobayar.getText());
            ep.setTgl_pembayaran(sdf.format(tgl.getDate()));
            ep.setNis(nis.getText());
            ep.setNip(nip.getText());
            ep.setBulan(j_bln.getText());
            ep.setJumlah(bayar1);
            btntambah.setText("Baru");

            if (ukem < nol || ubay < total){

```

```

        JOptionPane.showMessageDialog(null, "Uang
Bayar Kurang!!", "Informasi", JOptionPane.OK_OPTION);
        bayar.requestFocus();
        return;

    } else if (sp.edit(ep) == 1) {
        JOptionPane.showMessageDialog(null, "Simpan Data
Berhasil");
        cetak();
    } else {

    }

    BuatTabel();
    tombolAwal();
    Bersih();
    bayar.setText("0");
}

private void btnsimpanActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:
    if (nobayar.getText().isEmpty() || nis.getText().isEmpty()
        || jumbay.getText().isEmpty() || bulan.getSelectedIndex() ==
0) {
        JOptionPane.showMessageDialog(null, "Silahkan Lengkapi
Data !!");
    } else {

        total = Integer.valueOf(j_bln.getText()) + 1;

        j_bln.setText("" + total);
        int jml = (total * Integer.valueOf(jumbay.getText()));
        totalbayar.setText(String.valueOf(jml));
        ep.setId_pembayaran(nobayar.getText());
        ep.setTgl_pembayaran(sdf.format(tgl.getDate()));
        ep.setNis(nis.getText());
        ep.setNip(nip.getText());
        ep.setBulan(j_bln.getText());
        ep.setJumlah(Integer.parseInt(jumbay.getText()));
        et.setBayar(jumbay.getText());
    }
}

```

```

et.setBulan(bulan.getSelectedItem().toString());
et.setId_pembayaran(nobayar.getText());
et.setNis(nis.getText());

if (btntambah.getText().equalsIgnoreCase("Cetak"))
{ if (tbltrans.getRowCount() == 0) {
    if (sp.tambah(ep) == 1) {
        tm.tambah(et);
        entitas.nunggak n = new nunggak();
        method.Nunggak_Method mn = new
Nunggak_Method();
        mn.ubahlunas(nis.getText(),
bulan.getSelectedItem().toString());
        JOptionPane.showMessageDialog(null, "Simpan Data
Berhasil");
    } else {
        JOptionPane.showMessageDialog(null, "Simpan Data
Gagal !!");
    }
    tampil();
} else {
    if (tm.tambah(et) == 1) {

        JOptionPane.showMessageDialog(null, "Simpan Data
Berhasil");
    } else {
        JOptionPane.showMessageDialog(null, "Simpan Data
Gagal !!");
    }
    tampil();
}

}
bayar.setText("0");
bayar.requestFocus();
btntambah.setEnabled(true);
KurangBulan();
bulan.setSelectedIndex(0);

}
}

```

```

private void btnhapusActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here: total
    = Integer.valueOf(j_bln.getText()) - 1;

    j_bln.setText("" + total);
    int jml = (total * Integer.valueOf(jumbay.getText()));
    totalbayar.setText(String.valueOf(jml));
    et.setBulan(Bln);
    et.setNis(nis.getText());
    et.setId_pembayaran(nobayar.getText());

    int pesan = JOptionPane.showConfirmDialog(null, "Apakah Anda
Yakin Ingin Menghapus Data Ini ??", "Konfirmasi",
JOptionPane.YES_NO_OPTION,
JOptionPane.QUESTION_MESSAGE);
    if (pesan == JOptionPane.YES_OPTION) {
        if (tbltrans.getRowCount() == 1) {
            if ((sp.hapus(et) == 1) && (tm.hapus(et) == 1)) {
                cekbulan();
                entitas.nunggak n = new nunggak();
                method.Nunggak_Method mn = new
                Nunggak_Method(); n.setNis(nis.getText());
                JOptionPane.showMessageDialog(null, "Hapus Data
Berhasil");
                tampil();
            } else {
                JOptionPane.showMessageDialog(null, "Hapus Data Gagal
!!");
            }
        } else {
            if (sp.hapus(et) == 1) {
                cekbulan();
                JOptionPane.showMessageDialog(null, "Hapus Data
Berhasil");
                tampil();
            } else {
                JOptionPane.showMessageDialog(null, "Hapus Data Gagal
!!!");
            }
        }
    }
}

```

```

        }
    }
    SiapIsi(true);
}

private void btnkeluarActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:
    dispose();
}

private void tblsiswaMouseClicked(java.awt.event.MouseEvent evt)
{
    // TODO add your handling code here: int baris =
tblsiswa.getSelectedRow();
    nis.setText(tblsiswa.getModel().getValueAt(baris, 0).toString());
    txtnama.setText(tblsiswa.getModel().getValueAt(baris,
1).toString());
    kelas.setText(tblsiswa.getModel().getValueAt(baris, 2).toString());
    TA.setText(tblsiswa.getModel().getValueAt(baris, 3).toString());

    listSiswa = sm.tampilSiswa(nis.getText());
    kelas.setText(listSiswa.get(0).getKelas());
    TA.setText(listSiswa.get(0).getTahun_ajaran());
    jumbay.setText("'" + listSiswa.get(0).getB_spp());
    jDialog1.setVisible(false);
    cekbulan();
}

private void cariKeyReleased(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    String find = null;
    find = cari.getText();
    model.getDataVector().removeAllElements();
    model.fireTableDataChanged();
    listSiswa.clear();
    listSiswa = sm.cari(find);
    for (int i = 0; i < listSiswa.size(); i++)
    { Object[] a = new Object[4];
      a[0] = listSiswa.get(i).getNis();
      a[1] = listSiswa.get(i).getNama_siswa();
    }
}

```

```

        a[2] = listSiswa.get(i).getAlamat();
        a[3] = listSiswa.get(i).getTelephone();
        model.addRow(a);
    }
}

private void edcariActionPerformed(java.awt.event.ActionEvent evt)
{
    // TODO add your handling code here:
    jDialog1.pack();
    jDialog1.setAlwaysOnTop(true);
    jDialog1.setVisible(true);
    BuatTabelsiswa();
    tampilsiswa();
}

private void bayarKeyReleased(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if (bayar.getText().isEmpty()) {
        return;
    } else {
        int ubay = Integer.parseInt(bayar.getText());
        int grand = Integer.parseInt(totalbayar.getText());
        long ukem = ubay - grand;
        kembali.setText(Long.valueOf(ukem).toString());
        btntambah.setText("Cetak");
        return;
    }
}

private void tbltransMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    int baris = tbltrans.getSelectedRow();
    Bln = (tbltrans.getModel().getValueAt(baris,
    0).toString()); btnhapus.setEnabled(true);
    btnsimpan.setEnabled(false); btntambah.setEnabled(false);

    jLabel15.setText(Bln);
}

private void jScrollPane2MouseClicked(java.awt.event.MouseEvent

```

```

    evt) {
        // TODO add your handling code here:
    }

    private void btnbatalActionPerformed(java.awt.event.ActionEvent
    evt) {
        // TODO add your handling code here:
        Bersih();
        tombolAwal();
        cari.setEnabled(false);
        btnkeluar.setEnabled(true);
        btntambah.setEnabled(true);
        btnbatal.setEnabled(false);
        btntambah.setText("Baru");
    }

    private void bulanActionPerformed(java.awt.event.ActionEvent evt)
    {
        // TODO add your handling code here:
    }

    public static void main(String args[]) {

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Pembayaran().setVisible(true);
            }
        });
    }

    private void BuatTabelsiswa() {
        model = new DefaultTableModel();
        model.addColumn("NIS");
        model.addColumn("Nama");
        model.addColumn("Kelas");
        model.addColumn("Tahun Ajaran");
        tblsiswa.setModel(model);
    }

    private void tampilsiswa() {

```

```

        model.getDataVector().removeAllElements();
        model.fireTableDataChanged();
        listSiswa.clear();
        listSiswa = sm.tampil();
        for (int i = 0; i < listSiswa.size(); i++) {
            Object[] a = new Object[4];
            a[0] = listSiswa.get(i).getNis();
            a[1] = listSiswa.get(i).getNama_siswa();
            a[2] = listSiswa.get(i).getKelas();
            a[3] = listSiswa.get(i).getTahun_ajaran();
            model.addRow(a);
        }
    }

    private void buatNota() {
        try {

            String sql = "SELECT id_pembayaran FROM spp ORDER BY
id_pembayaran DESC";

            rs = st.executeQuery(sql);
            if (rs.next()) {
                String kd_barang =
rs.getString("id_pembayaran").substring(2);
                String AN = "" + (Integer.parseInt(kd_barang) +
1); String Nol = "";
                if (AN.length() == 1) {
                    Nol = "0000";
                } else if (AN.length() == 2)
                { Nol = "000";
                } else if (AN.length() == 3)
                { Nol = "00";
                } else if (AN.length() == 4)
                { Nol = "0";
                } else if (AN.length() == 5)
                { Nol = "";
                }
                nobayar.setText("N-" + Nol + AN);
            } else {
                nobayar.setText("N-00001");
            }
        }
    }

```

```

        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    private void tanggal(){
        Date now = new Date();
        SimpleDateFormat simpleDateFormat = new
SimpleDateFormat("dd");
        String tgl = simpleDateFormat.format(now);
        tanggal.setText(tgl);
    }

    private void tahun(){
        Date now = new Date();
        SimpleDateFormat simpleDateFormat = new
SimpleDateFormat("yy");
        String tgl = simpleDateFormat.format(now);
        tahun.setText(tgl);
    }

    private void bulan(){
        Date now = new Date();
        SimpleDateFormat simpleDateFormat = new
SimpleDateFormat("MM");
        String tgl = simpleDateFormat.format(now);
        bln.setText(tgl);
    }

    private void buatkode() {

        String thn = tahun.getText().substring(0, 2);
        String bulan= bln.getText().substring(0, 2);
        String tgl = tanggal.getText().substring(0, 2);
        String kode = (thn.substring(0,
2)+bulan.substring(0,2)+tgl.substring(0,2));
        try {
            //      kon.getKoneksi();
            String sql = "Select max(right(id_pembayaran,4)) as no From
spp";

```

```

rs = st.executeQuery(sql);
while (rs.next()) {
    if (rs.first() == false) {
        nobayar.setText(kode);
    } else {
        rs.last();
        int auto_id = rs.getInt(1) + 1;
        String no = String.valueOf(auto_id);
        int noLong = no.length();
        for (int a = 0; a < 4 - noLong; a++) {
            no = "0" + no;
        }
        nobayar.setText(kode+ no);
        String baca="select*from spp where id_pembayaran
In(Select Max(id_pembayaran)from spp) order by id_pembayaran
desc";

        rs = st.executeQuery(baca);
        while(rs.next())
        {
            txtambilno.setText(rs.getString("id_pembayaran"));
            String ambil = txtambilno.getText().substring(6,10);

            int ambil2 = Integer.parseInt(ambil);
            if(ambil2 == 9999){
                int auto = 000 + 1;
                String nomor = String.valueOf(auto);
                int noLong2 = nomor.length();
                for(int b=0;b<4-noLong2;b++){
                    nomor= "0"+nomor;
                }
                nobayar.setText(kode+nomor);

            }else if(ambil2 > 0 && ambil2 < 9999){
                int auto = ambil2+1;
                String nomor = String.valueOf(auto);
                int noLong2 = nomor.length();
                for(int b=0;b<4-noLong2;b++){
                    nomor= "0"+nomor;
                }
                nobayar.setText(kode+nomor);
            }
        }
    }
}

```

```

        }
    }
}
rs.close();
} catch (Exception e) {
    e.printStackTrace();
}
}

```

```

private void tombolAwal() {
    btnsimpan.setEnabled(false);
    btnhapus.setEnabled(false);
    btnkeluar.setEnabled(true);
}

```

```

private void Bersih() {
    nobayar.setText(null);
    nis.setText(null);
    bulan.setSelectedIndex(0);
    totalbayar.setText("0");
    kembali.setText("0");
    j_bln.setText("0");
    totalbayar.setText("0");
    txtnama.setText(null);
    kelas.setText(null);
    TA.setText(null);
    jumbay.setText(null);
    totalbayar.setText("0");
}

```

```

private void Bersih2() {

    bulan.setSelectedIndex(0);
    totalbayar.setText("0");
    j_bln.setText("0");
    totalbayar.setText("0");
    txtnama.setText(null);
    kelas.setText(null);
    TA.setText(null);
    jumbay.setText(null);
}

```

```

        totalbayar.setText("0");
        kembali.setText("0");
    }

    private void BuatTabel() {
        modeltrnas = new DefaultTableModel();
        modeltrnas.addColumn("Bulan");
        modeltrnas.addColumn("Bayar");
        tbltrans.setModel(modeltrnas);
    }

    private void tampil() {
        modeltrnas.getDataVector().removeAllElements();
        modeltrnas.fireTableDataChanged();
        listSpp.clear();
        listSpp = tm.tampilBayar(nobayar.getText(),
            nis.getText()); for (int i = 0; i < listSpp.size(); i++) {
            Object[] a = new Object[2];
            a[0] = listSpp.get(i).getBulan();
            a[1] = listSpp.get(i).getBayar();
            modeltrnas.addRow(a);
        }
    }

    private void SiapIsi(boolean a) {
        nobayar.setEnabled(false);
        nis.setEnabled(false);
        bulan.setEnabled(false);
        edcari.setEnabled(false);
        btnhapus.setEnabled(false);
        btnsimpan.setEnabled(false);
        btnbatal.setEnabled(false);
        tgl.setEnabled(false);
        nip.setEnabled(false);
        jumbay.setEnabled(false);
        txtnama.setEnabled(false);
        TA.setEnabled(false);
        kelas.setEnabled(false);
    }

    private void non() {

```

```

        nobayar.setEnabled(true);
        nis.setEnabled(true);
        bulan.setEnabled(true);
        edcari.setEnabled(true);
        btnhapus.setEnabled(true);
        btnsimpan.setEnabled(true);
        btnbatal.setEnabled(true);
        tgl.setEnabled(true);
        nip.setEnabled(true);
        jumbay.setEnabled(true);
        txtnama.setEnabled(true);
        TA.setEnabled(true);
        kelas.setEnabled(true);
    }

    void cekbulan() {
        bulan.setModel(new javax.swing.DefaultComboBoxModel(new
String[]{"pilih", "Juli", "Agustus", "September", "Oktober",
"November", "Desember", "Januari", "Februari", "Maret", "April",
"Mei", "Juni"}));
        listSpp = tm.tampilBulan(nis.getText());
        for (int i = 0; i < listSpp.size(); i++) {

            Object[] a = new Object[6];

            bulan.removeItem(a[1] = listSpp.get(i).getBulan());
        }
        if (bulan.getItemCount() == 1) {
            JOptionPane.showMessageDialog(null, "Siswa Telah Melunasi
SPP Tahun Ajaran Ini !!!");
            btnsimpan.setEnabled(false);
            btntambah.setEnabled(true);
            Bersih2();
        } else {
            btnsimpan.setEnabled(true);
        }
    }

    void KurangBulan() {
        bulan.setModel(new javax.swing.DefaultComboBoxModel(new
String[]{"pilih", "Juli", "Agustus", "September", "Oktober",

```

```

"November", "Desember", "Januari", "Februari", "Maret", "April",
"Mei", "Juni"}));
listSpp = tm.tampilBulan(nis.getText());
for (int i = 0; i < listSpp.size(); i++) {

    Object[] a = new Object[6];

    bulan.removeItem(a[1] = listSpp.get(i).getBulan());
}
if (bulan.getItemCount() == 1) {
    JOptionPane.showMessageDialog(null, "Siswa Telah Melunasi
SPP Tahun Ajaran Ini !!!");
    btnsimpan.setEnabled(false);
} else {
}
}

void cetak() {
    try {

        File file = new File("src/laporan/Kwitansi.jrxml");
        jasperDesign1 = JRXmlLoader.load(file);
        param.put("id", nobayar.getText());
        param.put("nip", Dashboard.nip.getText());
        jasperReport1 =
JasperCompileManager.compileReport(jasperDesign1);
        jasperPrint1 = JasperFillManager.fillReport(jasperReport1,
param, method.koneksi.getKoneksi());

        JasperViewer.viewReport(jasperPrint1, false);
    } catch (Exception e) {
        e.printStackTrace();
    }
}
}

```

4. **Source Code Ganti Password**

```

package view;
import static view.form_login.idtf;
import java.awt.event.KeyEvent;
import java.sql.SQLException;
import javax.swing.JOptionPane;

```

```

import javax.swing.JTextField;
import entitas.*;
import method.*;

/**
 *
 * @author
 * Asus */
public class GantiPass extends javax.swing.JFrame {
    koneksi kon = new koneksi();
    Entitas_Pass ep = new Entitas_Pass();

    public GantiPass() {
        initComponents();
        kon.getKoneksi();
        nip.setText(DashboardAdmin.nip.getText());
        nip.setText(Dashboard_kepsek.nip.getText());
        nip.setText(Dashboard.nip.getText());
        awal();
        nip.setEnabled(false);
        tlama.setEnabled(true);
        tlama.requestFocus();
        tbaru.setEnabled(false);
        tkonfirm.setEnabled(false);

    }

    void bersih()
    {
        tlama.setText("");
        tbaru.setText("");
        tkonfirm.setText("");
    }

    void aktif()
    {
        tlama.setEnabled(true);
    }

    void mati()

```

```

{
    nip.setEnabled(false);
    tlama.setEnabled(false);
    tbaru.setEnabled(false);
    tkonfirm.setEnabled(false);
}

void awal()
{
    try{
        String sql="select*from user where
        nip='"+nip.getText()+"''"; kon.rs = kon.st.executeQuery(sql);
        if(kon.rs.next()){
            pass.setText(kon.rs.getString("password"));
            tlama.setEnabled(true);
            tlama.requestFocus();
            nip.setEnabled(false);
        }
    }catch(SQLException e)
    {System.out.println("Koneksi Gagal"+ e.toString());}
    pass.setVisible(true);
    tlama.setText("");
    tbaru.setText("");
    tkonfirm.setText("");
    nip.setEnabled(false);
    tlama.setVisible(true);
    tlama.requestFocus();
}

@SuppressWarnings("unchecked")
private void bttutupActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    this.dispose();
}

private void btperbaruiActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:
    try{
        if(nip.getText().isEmpty() ||
        tlama.getText().isEmpty() ||

```

```

        tbaru.getText().isEmpty() ||
        tkonfirm.getText().isEmpty()){
            JOptionPane.showMessageDialog(this,"Lengkapi data",
            "Informasi",JOptionPane.OK_OPTION);
        }else{

            if(tkonfirm.getText().isEmpty() ||
            tbaru.getText().isEmpty()){ return;
            }
            String baru = String.valueOf(tbaru.getText());
            String konfirm = String.valueOf(tkonfirm.getText());

            if(String.valueOf(konfirm).equalsIgnoreCase(String.valueOf(baru))){
                tbaru.setEnabled(false);
                tkonfirm.setEnabled(false);

                String sql="update user set
                password='"+tkonfirm.getText()+
                ""+"' where nip ='"+nip.getText()+"'";
                kon.st.executeUpdate(sql);

                JOptionPane.showMessageDialog(null,"Password berhasil
                diperbarui");
                awal();
                bersih();
            }else if(String.valueOf(konfirm) != String.valueOf(baru)){
                JOptionPane.showMessageDialog(null,"Password tidak sama
                !", "Informasi",JOptionPane.OK_OPTION);
                tbaru.setText("");
                tkonfirm.setText("");
                tbaru.setEnabled(true);
                tbaru.requestFocus();
            }
        }
    } catch(SQLException e)
    {
    }
}

private void btbatalActionPerformed(java.awt.event.ActionEvent
evt) {

```

```

        // TODO add your handling code here:
        awal();
        aktif();
        bersih();
        tbaru.setEnabled(false);
        tkonfirm.setEnabled(false);
    }

    private void tbaruKeyPressed(java.awt.event.KeyEvent evt) {
        // TODO add your handling code here:
        if(evt.getKeyCode()==KeyEvent.VK_ENTER)
        {
            if(tbaru.getText().isEmpty())
            {
                return;
            }
            else
            {
                tkonfirm.setText("");
                tkonfirm.requestFocus();
            }
        }
    }

    private void tlamaKeyPressed(java.awt.event.KeyEvent evt) {
        // TODO add your handling code here:
        if(evt.getKeyCode()==KeyEvent.VK_ENTER)
        {
            if(tlama.getText().isEmpty())
            {
                return;
            }
            else
            {
                String pass1 = String.valueOf(pass.getText()); String
                pass2 = String.valueOf(tlama.getText());
                if(String.valueOf(pass2).equalsIgnoreCase(pass1)){
                    tlama.setEnabled(false);
                    tbaru.setEnabled(true);
                    tkonfirm.setEnabled(true);
                    tbaru.requestFocus();
                }
            }
        }
    }

```

```

        try{
            String sql="Select*from user where nip="+
n            nip.getText() +
                "" and password="" + tlama.getText() +""";
            kon.rs = kon.st.executeQuery(sql);

        }catch(Exception e){
            return;
        }
    }
    else if(String.valueOf(pass2) != String.valueOf(pass1)){
        JOptionPane.showMessageDialog(null,"Password tidak
sesuai !!", "Informasi",JOptionPane.OK_OPTION);
        tlama.setText("");
        tbaru.requestFocus();
    }

}
}
}

private void tconfirmKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if(evt.getKeyCode()==KeyEvent.VK_ENTER){
        if(tbaru.getText().isEmpty()){
            return;
        }else
        {return;}
    }
}

private void tlamaKeyReleased(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if(tlama.getText().isEmpty()){
        return;
    }
}

private void tbaruKeyReleased(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if(tbaru.getText().isEmpty()){

```

```

        return;
    }
}

private void tkonfirmKeyReleased(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if(tkonfirm.getText().isEmpty()){
        return;
    }
}

private void formWindowOpened(java.awt.event.WindowEvent evt)
{
    // TODO add your handling code here:
    awal();
}

private void nipKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if(evt.getKeyCode()==KeyEvent.VK_ENTER){
        if(nip.getText().isEmpty())
            {return;}

        else{
            try
            {
                String sql="select*from user where
nip='"+nip.getText()+"'";
                kon.rs = kon.st.executeQuery(sql);
                if(kon.rs.next())
                {
                    pass.setText(kon.rs.getString("password"));
                    tlama.setEnabled(true);
                    tlama.requestFocus();
                    nip.setEnabled(false);
                }
            }
            else
            {
                JOptionPane.showMessageDialog(null,"Username
Salah !!", "Informasi",
                JOptionPane.OK_OPTION);
            }
        }
    }
}

```

```

        nip.setText("");
        nip.requestFocus();
    }
}
catch(SQLException e)
{
    System.out.println("Koneksi Gagal"+ e.toString());
}
}
}
}

public static void main(String args[]) {
    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new GantiPass().setVisible(true);
        }
    });
}
}

```

5. **Source Code Laporan**

```

package view;

import java.io.File; import
java.util.HashMap; import
java.util.Map;
import net.sf.jasperreports.engine.JasperCompileManager;
import net.sf.jasperreports.engine.JasperFillManager; import
net.sf.jasperreports.engine.JasperPrint;
import net.sf.jasperreports.engine.JasperReport; import
net.sf.jasperreports.engine.design.JasperDesign; import
net.sf.jasperreports.engine.xml.JRXmlLoader; import
net.sf.jasperreports.view.JasperViewer;

/**
 *
 * @author
 * Asus */
public class Lap_Kelas extends javax.swing.JFrame
{
    JasperReport jasperReport1;
    JasperDesign jasperDesign1;
}

```

```

JasperPrint jasperPrint1;

Map<String, Object> param = new HashMap<String, Object>();

public Lap_Kelas() {
    initComponents();
    setLocationRelativeTo(this);
}

@SuppressWarnings("unchecked")
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try{

        File file = new File("src/laporan/LaporanDataKelas.jrxml");
        jasperDesign1 = JRXmlLoader.load(file);
        param.put("nip",Dashboard.nip.getText());
        jasperReport1 =
JasperCompileManager.compileReport(jasperDesign1);
        jasperPrint1 = JasperFillManager.fillReport(jasperReport1,
param,method.koneksi.getKoneksi());

        JasperViewer.viewReport(jasperPrint1,false);
    } catch (Exception e) {
        e.printStackTrace();
    }
}

public static void main(String args[ ]) {
    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable()
    { public void run() {
        new Lap_Kelas().setVisible(true);
    }
    });
}

```