

Perancangan Sistem Aplikasi Pemesanan Lapangan Futsal Berbasis Java GUI

Dokumen ini ditujukan untuk memenuhi Tugas Akhir (UAS) Mata Kuliah

Pemrograman Berbasis Obyek Dosen Pengampu Ibu Ratna Mufidah , S.Kom., M.Kom.



Kelas : 4A

Disusun Oleh :

Alpin Apriliansyah Mohsa (2010631250003)

Azki Nurul Fajri (2010631250005)

Ade Sutrisna Darsanjaya (2010631250084)

Dias Raihan Azahri (2010631250007)

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SINGAPERBANGSA KARAWANG**

2022

Daftar Isi

BAB I PENDAHULUAN.....	4
1.1 Latar Belakang	4
1.2 Identifikasi Permasalahan	4
1.3 Rumusan Masalah	4
1.4 Tujuan dan Manfaat Penelitian	4
1.4.1 Tujuan.....	4
1.4.2 Manfaat.....	5
BAB II PEMBAHASAN dan HASIL	6
2.1 Class Diagram	6
2.2 Database Diagram	6
2.3 Design Gui.....	7
2.4 Penjelasan fitur-fitur yang ada dalam GUI	10
2.5 Kode Program	12
2.6 Screenshot	31

Tabel Pembagian Tugas Anggota Kelompok

Nama Anggota	Pembagian Kerja
Alpin Apriliansyah Mohsa	Pembuatan Source Code kerja sistem
Azki Nurul Fajri	Pembuatan Source Code kerja sistem
Ade Sutrisna Darsanjaya	Pembuatan Database sistem
Dias Raihan Azahri	Pembuatan Design GUI

BAB I

PENDAHULUAN

1.1 Latar Belakang

Futsal adalah salah satu jenis olahraga yang sedang populer saat ini. Seiring dengan berkembangnya olahraga futsal, ikut berkembang juga peluang bisnis penyewaan lapangan futsal untuk memenuhi kebutuhan konsumen yang ingin bermain futsal akan tetapi tidak mempunyai lapangan untuk bermain. Menurut Journal of Sport Sciences and Fitness (2013:15) “futsal adalah kata yang digunakan secara internasional untuk permainan sepakbola dalam ruangan. kata itu berasal dari kata FUTbol atau FUTebol (dari bahasa Spanyol atau Portugal yang berarti permainan sepakbola) dan Salon atau SALa(dari bahasa Prancis atau Spanyol yang berarti dalam ruangan). Perbedaan antara futsal dengan sepak bola ada pada ukuran lapangan nya lapangan futsal lebih kecil dari sepak bola biasa. Jumlah pemain nya pun hanya 5 orang dan biasanya dimainkan di dalam ruangan atau tertutup walaupun ada yang dibuat di area terbuka.

Pada zaman sekarang, perkembangan teknologi informasi memegang peranan penting dalam dunia usaha. Tidak sedikit dari usaha-usaha dalam skala besar, sedang, maupun kecil yang saat ini sudah memanfaatkan teknologi informasi untuk menunjang usahanya. Salah satu teknologi informasi yang saat ini sangat dibutuhkan dan sering digunakan adalah teknologi informasi berupa aplikasi.

Salah satu usaha yang saat ini masih minim menggunakan teknologi informasi adalah usaha penyewaan lapangan futsal. Dimana saat ini banyak penyewaan lapangan futsal yang masih menggunakan cara manual untuk menunjang usahanya itu, contohnya untuk melakukan booking lapangan, pegawai lapangan futsal masih melakukan pencatatan dengan manual, cara tersebut kurang efektif dan efisien.

1.2 Identifikasi Permasalahan

Berdasarkan latar belakang diatas, Penulis mengidentifikasi permasalahan yang ada pada Latar Belakang tersebut. Adapun identifikasi permasalahan ini adalah sebagai berikut :

1. Pencatatan penyewaan lapangan Futsal Kipindeas masih secara manual.
2. Kurang nya pemanfaat Teknologi pada era sekarang.
3. Pencatatan keuangan penyewaan lapangan Futsal Kipindeas masih secara Manual.

1.3 Rumusan Masalah

Berdasarkan latar belakang di atas, maka Rumusan Masalah dalam penelitian ini adalah “”Bagaimana mempermudah dalam melakukan pencatatan keuangan maupun pencatatan penyewaan Lapangan Futsal Kipindeas”

1.4 Tujuan dan Manfaat Penelitian

1.4.1 Tujuan

Adapun tujuan dari penelitian ini adalah untuk membuat Aplikasi Berbasis Object Oriented Pemesanan Lapangan Futsal Kipindeas guna membantu dan mempermudah user dalam melakukan pemesanan lapangan futsal Kipindeas.

1.4.2 Manfaat

Adapun manfaat dalam penelitian ini adalah sebagai berikut :

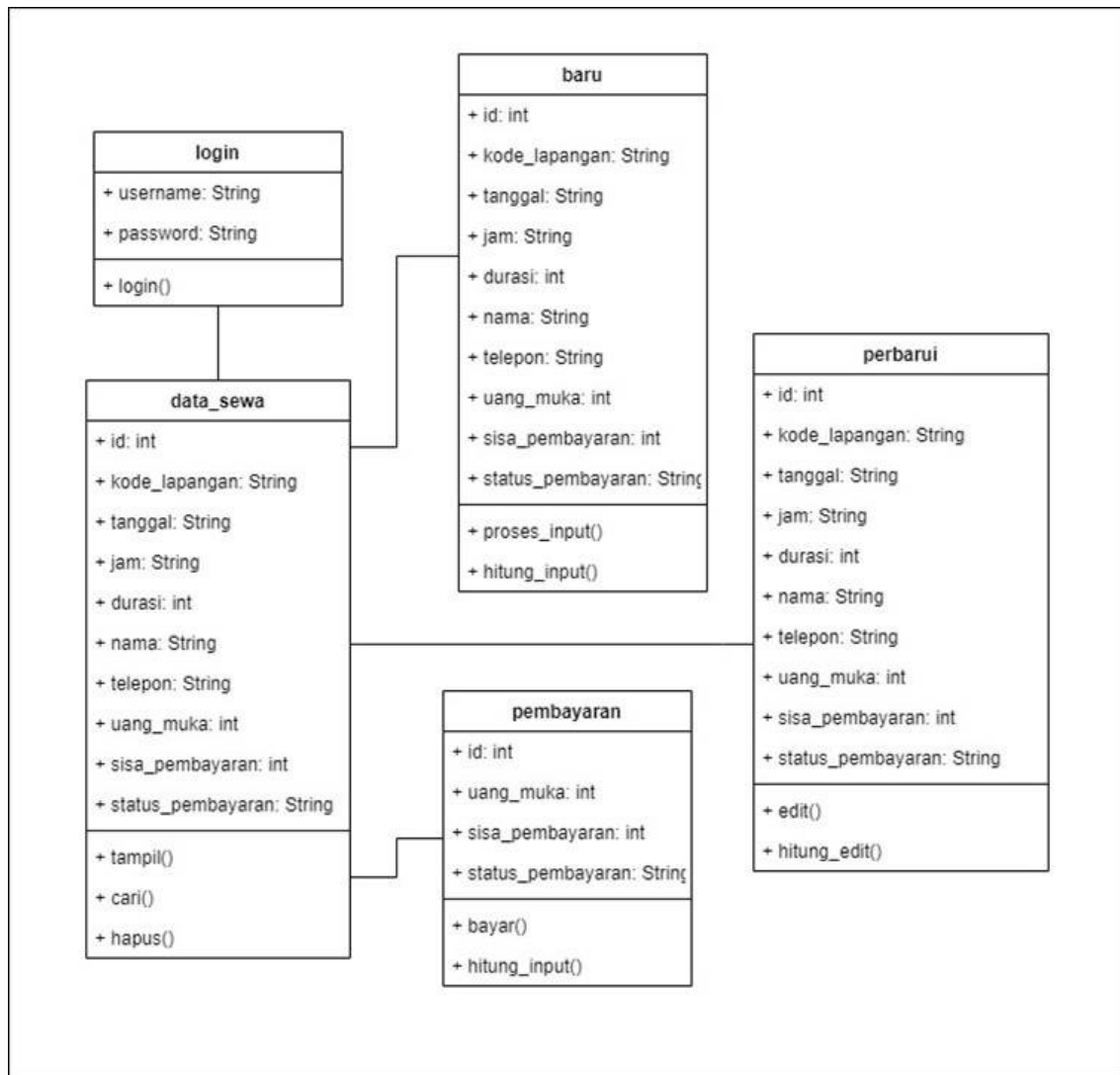
- a. Dapat mempermudah kasir untuk melihat daftar jadwal yang telah dipesan.
- b. Dapat mengetahui kondisi Lapangan, apakah masih *ready* atau tidak.
- c. Dapat membantu pemesan untuk memesan lapangan futsal.
- d. Dapat membantu kasir untuk melihat pembayaran.

BAB II

PEMBAHASAN dan HASIL

2.1 Class Diagram

Pada aplikasi penyewaan Lapangan Futsal Kipindeas ini memiliki *Class Diagram* yang terdiri dari Class bari, Class Login, Class perbarui, dan Class Pembayaran. Dari Class-Class yang ada pada Aplikasi Penyewaan Lapangan Futsal Kipindeas ini memiliki variable-variable dan Method. Berikut lampiran Class Diagram Pada Aplikasi Penyewaan Lapangan Futsal Kipindeas.



2.2 Database Diagram

Database diagram pada Aplikasi Penyewaan Lapangan Futsal Kipindeas ini memiliki 1 Database yang Bernama database_futsal dan memiliki 2 Table yaitu Table data_sewa dan Table login. Pada Table data_sewa memiliki beberapa variable dan masing-masing memiliki fungsinya sendiri, pada table login ini berfungsi untuk Class login dan memiliki beberapa variable yang masing-masing mempunyai fungsinya.

database: futsal data_sewa	database: futsal login
tbl_id : int(4)	username : varchar(15)
tbl_kode_lapangan : varchar(6)	password : varchar(10)
tbl_tanggal : varchar(20)	
tbl_jam : varchar(15)	
tbl_durasi : varchar(10)	
tbl_nama : varchar(50)	
tbl_telepon : varchar(13)	
tbl_uang_muka : varchar(10)	
tbl_sisa_pembayaran : varchar(10)	
tbl_status_pembayaran : varchar(20)	

2.3 Design Gui

Sebelum membuat aplikasi penyewaan Lapangan Kipindeas, penulis membuat Design Gui untuk merancang Aplikasi. Pada Aplikasi Penyewaan Lapangan Kipindeas ini memiliki Design Gui yang terdiri design Login, design Data Sewa, design Baru, design Perbarui, dan design Pembayaran. Berikut lampiran design-design GUI pada Aplikasi Penyewaan Lapangan Futsal Kipindeas.

a. Design Login

LOGIN ADMIN

Nama User

Kata sandi

b. Design Data Sewa

DATA BOOKING										
ID Sewa	Kode Lapangan	Tanggal	Jam	Durasi	Nama	No. HP	Uang Muka	Sisa Bayar	Status	

Copyright by: KIPINDEAS

c. Design Data Baru

DATA BARU	
ID Sewa	<input type="text"/>
Kode Lapangan	<input type="text"/>
Tanggal Sewa	<input type="text"/> <input type="text"/> <input type="text"/>
Jam Sewa	<input type="text"/> Sampai <input type="text"/>
Durasi (Jam)	<input type="text"/>
Nama	<input type="text"/>
No HP	<input type="text"/>
Uang Muka	<input type="text"/>
Sisa Bayar	<input type="text"/> <input type="button" value="Hitung"/>
Status	<input type="text"/>
<input type="button" value="Batal"/> <input type="button" value="Proses"/>	

d. Design Perbarui Data

PEMBARUAN DATA

ID Sewa

Kode Lapangan

Tanggal Sewa

Jam Sewa

Durasi (Jam)

Nama

No HP

Uang Muka

Sisa Bayar

Hitung

Status

Batal

Perbarui

e. Design Pembayaran

PEMBAYARAN

ID

Uang Masuk

Sisa Bayar

Hitung

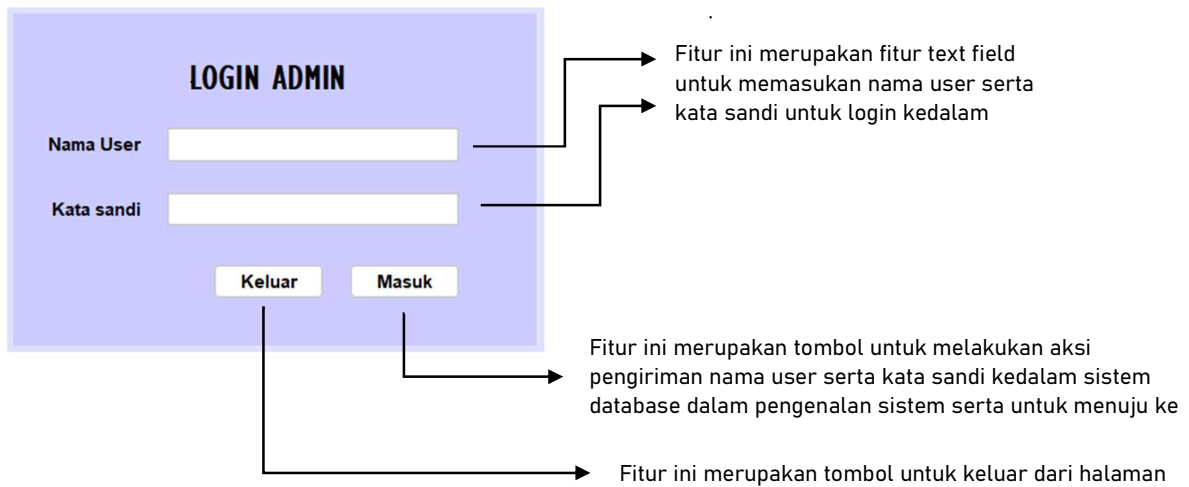
Status

Batal

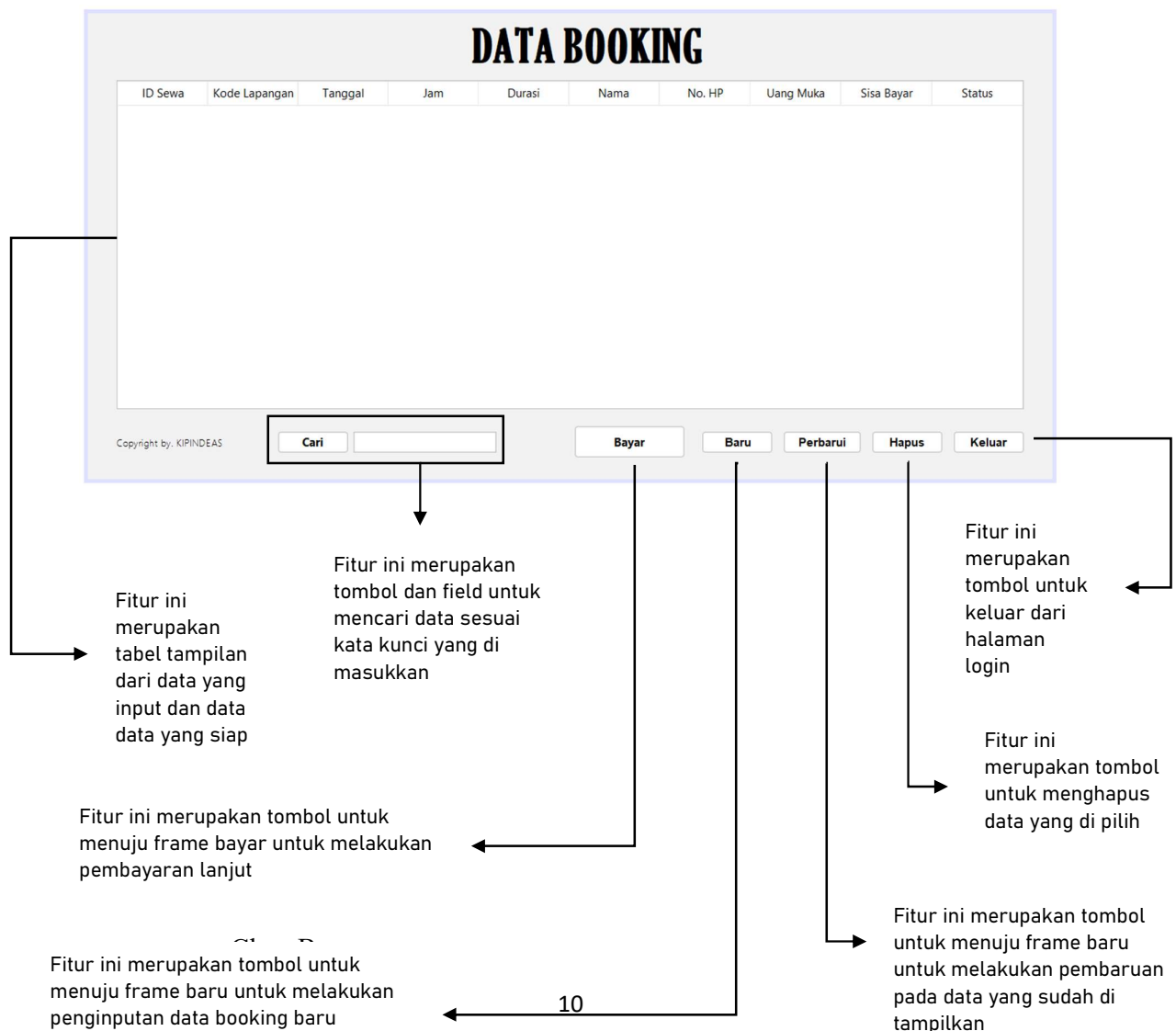
Bayar

2.4 Penjelasan fitur-fitur yang ada dalam GUI

Class Login



Class Data Sewa



DATA BARU

ID Sewa

Kode Lapangan

Tanggal Sewa

Jam Sewa Sampai

Durasi (Jam)

Nama

No HP

Uang Muka

Sisa Bayar

Status

Fitur ini merupakan fitur penginputan data baru yang terdiri dari textfield dan juga beberapa combo box, disini ada 1 textfield yang tidak bisa diedit yaitu pada textfield "Sisa Bayar" yang digunakan untuk menampilkan hasil hitung pada tombol hitung

Fitur ini merupakan tombol yang berfungsi untuk melakukan penghitungan otomatis yang menunjukkan hasil Sisa Pembayaran dari pemesanan

Fitur ini merupakan tombol yang berfungsi untuk melakukan proses penginputan data yang akan di tampilkan pada halaman utama data booking

Fitur ini merupakan tombol yang berfungsi untuk membatalkan aksi penginputan dan langsung menuju halaman data booking

Class Perbarui

PEMBARUAN DATA

ID Sewa

Kode Lapangan

Tanggal Sewa

Jam Sewa

Durasi (Jam)

Nama

No HP

Uang Muka

Sisa Bayar

Status

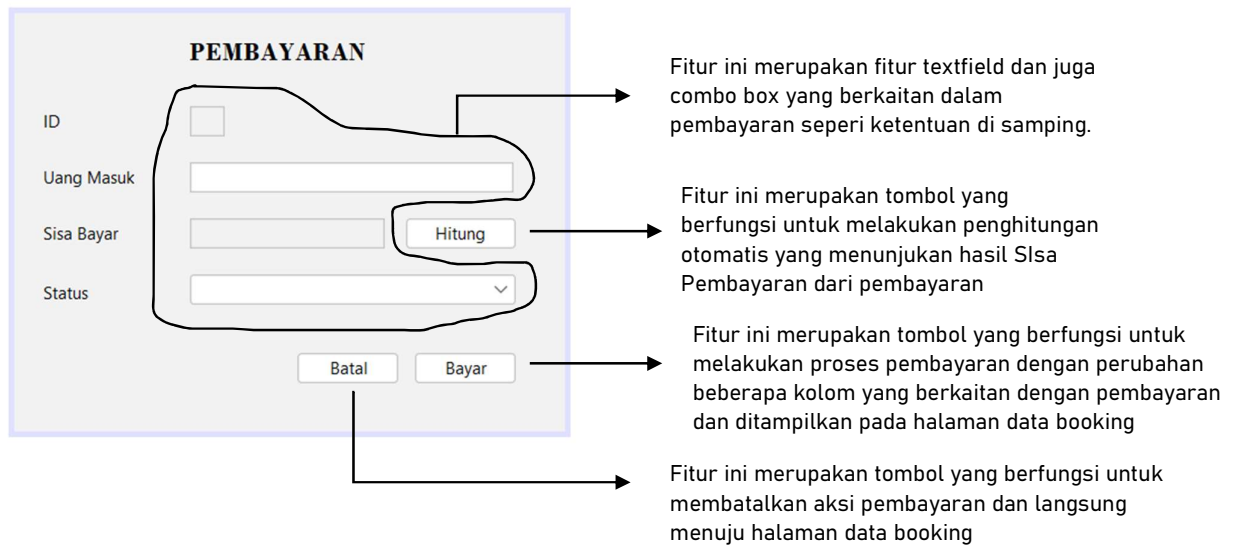
Fitur ini merupakan fitur yang hamper sama dengan yang sama yang ada pada fitur penginputan class Baru. Akan tetapi, data yang sebelumnya sudah di inputkan akan di tampilkan kembali untuk di ubah jika ada perubahan data.

Fitur ini merupakan tombol yang berfungsi untuk melakukan penghitungan otomatis yang menunjukkan hasil Sisa Pembayaran dari pemesanan

Fitur ini merupakan tombol yang berfungsi untuk melakukan proses pembaruan data yang akan di tampilkan pada halaman utama data booking

Fitur ini merupakan tombol yang berfungsi untuk membatalkan aksi pembaruan dan langsung menuju halaman data booking

Class Pembayaran



2.5 Kode Program

Dalam pembuatan Kode Program sekiranya ada 6 Class + 1 database yang dibuat dalam pembuatan sistem ini dengan kode program sebagai berikut.

- Kelas koneksi.java yang berfungsi untuk menyambungkan database MySQL kedalam java

Baris	koneksi.java
1	import java.sql.DriverManager;
2	import java.sql.Connection;
3	import java.sql.Statement;
4	import java.sql.ResultSet;
5	import java.sql.SQLException;
6	/**
7	*
8	* @author USER
9	*/
10	public class koneksi {
11	//jdbc:jenisdatabaseserver://alamatdatabase:portdatabase//namadatabase
12	static final String JDBC_URL ="jdbc:mysql://localhost:3306/database_futsal";
13	static final String USER = "root";
14	static final String PASSWORD = "";
15	static Connection connection;
16	static Statement statement;
17	public void config(){
18	try {
19	connection = DriverManager.getConnection(JDBC_URL,USER,PASSWORD);
20	statement = connection.createStatement();
	} catch (SQLException e) {

21	System.out.println(e);
22	}
23	}
24	}

- Kelas login.java (JFrame login)

Baris	login.java
1	import java.sql.Connection;
2	import java.sql.ResultSet;
3	import java.sql.Statement;
4	import javax.swing.JOptionPane;
5	import java.sql.SQLException;
6	import java.sql.DriverManager;
7	import java.util.logging.Level;
8	import java.util.logging.Logger;
9	
10	/**
11	*
12	* @author USER
13	*/
14	public class login extends javax.swing.JFrame {
15	
16	static Connection conn;
17	static ResultSet res;
18	static Statement state;
19	
20	public login() throws SQLException {
21	initComponents();
22	
23	koneksi DB = new koneksi();
24	DB.config();
25	
26	conn = DB.connection;
27	state = DB.statement;
28	
29	}
30	
31	private void login(){
32	try {
33	String sql = "SELECT * FROM login WHERE username='"+txt_username.getText()
34	+ "' AND password='"+psf_password.getText()+"'";
35	res = state.executeQuery(sql);
36	if(res.next()){
37	if(txt_username.getText().equals(res.getString("username"))
38	&& psf_password.getText().equals(res.getString("password"))){
	JOptionPane.showMessageDialog(null, "berhasil login");

```

39
40         data_sewa gui = new data_sewa();
41         gui.setVisible(true);
42         this.dispose();
43     }
44 } else {
45     JOptionPane.showMessageDialog(null, "username atau password salah");
46     txt_username.requestFocus();
47 }
48 } catch (Exception e) {
49     JOptionPane.showMessageDialog(this, e.getMessage());
50 }
51 }
52
53 /**
54  * This method is called from within the constructor to initialize the form.
55  * WARNING: Do NOT modify this code. The content of this method is always
56  * regenerated by the Form Editor.
57  */
58 @SuppressWarnings("unchecked")
59 Generated Code
190
191 private void psf_passwordActionPerformed(java.awt.event.ActionEvent evt) {
192     // TODO add your handling code here:
193 }
194
195 private void btn_masukActionPerformed(java.awt.event.ActionEvent evt) {
196     login();
197 }
198
199 private void btn_keluarActionPerformed(java.awt.event.ActionEvent evt) {
200     int jawab = JOptionPane.showConfirmDialog(
201         null, "Apakah anda ingin keluar dari Program?", "Keluar Program",
202         JOptionPane.YES_NO_OPTION);
203
204     if (jawab == JOptionPane.YES_OPTION) {
205         System.exit(0);
206     } else {
207         txt_username.requestFocus();
208     }
209 }
210
211 /**
212  * @param args the command line arguments
213  */
214 public static void main(String args[]) {
215     /* Set the Nimbus look and feel */
216     Look and feel setting code (optional)

```

237	
238	/* Create and display the form */
239	java.awt.EventQueue.invokeLater(new Runnable() {
240	public void run() {
241	try {
242	new login().setVisible(true);
243	} catch (SQLException ex) {
244	Logger.getLogger(login.class.getName()).log(Level.SEVERE, null, ex);
245	}
246	}
247	});
248	}
249	
250	// Variables declaration - do not modify
251	private javax.swing.JButton btn_keluar;
252	private javax.swing.JButton btn_masuk;
253	private javax.swing.JButton jButton1;
254	private javax.swing.JLabel jLabel1;
255	private javax.swing.JPanel jPanel1;
256	private javax.swing.JPanel jPanel2;
257	private javax.swing.JLabel lbl_Login;
258	private javax.swing.JLabel lbl_password;
259	private javax.swing.JLabel lbl_username;
260	private javax.swing.JPasswordField psf_password;
261	private javax.swing.JTextField txt_username;
262	// End of variables declaration
263	}

- Kelas data_sewa (JFrame Data Booking/Halaman Utama)

Baris	data_sewa.java
1	import javax.swing.table.DefaultTableModel;
2	import java.awt.HeadlessException;
3	import java.sql.Connection;
4	import java.sql.ResultSet;
5	import java.sql.Statement;
6	import javax.swing.JOptionPane;
7	import java.sql.SQLException;
8	import java.sql.PreparedStatement;
9	
10	/**
11	*
12	* @author USER
13	*/
14	public class data_sewa extends javax.swing.JFrame {
15	
16	static Connection conn;
17	static ResultSet res;
18	static Statement state;

```

19
20 public data_sewa() {
21     initComponents();
22     //koneksi ke database
23     koneksi DB = new koneksi();
24     DB.config();
25     conn = DB.connection;
26     state = DB.statement;
27     tampil();
28 }
29
30 public void tampil(){
31     try{
32         Object [] rows={"ID","Kode Lapangan","Tanggal","Jam","Durasi","Nama","No. HP","Uang
33 Muka","Sisa Bayar","Status"};
34         DefaultTableModel dtm = new DefaultTableModel(null,rows);
35         tbl_data.setModel(dtm);
36         tbl_data.setBorder(null);
37
38         String tbl_id="",tbl_kode_lapangan="",tbl_tanggal="",tbl_jam="",tbl_durasi="",tbl_nama=""
39 +
40 "",tbl_telepon="",tbl_uang_muka="",tbl_sisa_pembayaran="",tbl_status_pembayaran="";
41         try{
42             String sql="SELECT * FROM data_sewa";
43             state=conn.createStatement();
44             res=state.executeQuery(sql);
45             while(res.next()){
46                 tbl_id=res.getString("tbl_id");
47                 tbl_kode_lapangan=res.getString("tbl_kode_lapangan");
48                 tbl_tanggal=res.getString("tbl_tanggal");
49                 tbl_jam=res.getString("tbl_jam");
50                 tbl_durasi=res.getString("tbl_durasi");
51                 tbl_nama=res.getString("tbl_nama");
52                 tbl_telepon=res.getString("tbl_telepon");
53                 tbl_uang_muka=res.getString("tbl_uang_muka");
54                 tbl_sisa_pembayaran=res.getString("tbl_sisa_pembayaran");
55                 tbl_status_pembayaran=res.getString("tbl_status_pembayaran");
56                 String [] tampil={" "+tbl_id,tbl_kode_lapangan,tbl_tanggal,tbl_jam,tbl_durasi,tbl_nama
57 ,tbl_telepon,tbl_uang_muka,tbl_sisa_pembayaran,tbl_status_pembayaran};
58                 dtm.addRow(tampil);
59             }
60         } catch(SQLException e){
61             e.printStackTrace();
62             JOptionPane.showMessageDialog(null,"Query Salah "+e);
63         }
64     } catch(Exception e){
65         e.printStackTrace();
66     }

```



```

67     }
68
69     public void hapus(){
70         int ok=JOptionPane.showConfirmDialog(null,"Apakah Yakin Mendelete Record Ini???",
71             "Confirmation",JOptionPane.YES_NO_OPTION);
72         if (ok==0){
73             try{
74                 String sql="delete from data_sewa where
75 tbl_id='"+tbl_data.getValueAt(tbl_data.getSelectedRow(), 0)+"''";
76                 PreparedStatement state=conn.prepareStatement(sql);
77                 state.executeUpdate();
78                 tampil();
79                 JOptionPane.showMessageDialog(null, "Delete Data Sukses");
80             }
81             catch (Exception e){
82                 JOptionPane.showMessageDialog(null, "Delete Data Gagal");
83             }
84         }
85     }
86
87     public void cari(){
88         Object [] rows={"ID","Kode Lapangan","Tanggal","Jam","Durasi","Nama","No. HP","Uang
89 Muka","Sisa Bayar","Status"};
90         DefaultTableModel dtm = new DefaultTableModel(null,rows);
91         tbl_data.setModel(dtm);
92         tbl_data.setBorder(null);
93
94
95         String tbl_id="",tbl_kode_lapangan="",tbl_tanggal="",tbl_jam="",tbl_durasi="",tbl_nama=""
96         +
97         "",tbl_telepon="",tbl_uang_muka="",tbl_sisa_pembayaran="",tbl_status_pembayaran="";
98
99         String cari = txt_cari.getText();
100
101         try{
102             String sql="SELECT * FROM data_sewa WHERE tbl_id LIKE '%" +cari+"%'
103                 + "OR tbl_kode_lapangan LIKE '%" +cari+"%' OR tbl_tanggal LIKE '%" +cari+"%'
104                 + "OR tbl_jam LIKE '%" +cari+"%' OR tbl_durasi LIKE '%" +cari+"%'
105                 + "OR tbl_nama LIKE '%" +cari+"%' OR tbl_telepon LIKE '%" +cari+"%'
106                 + "OR tbl_uang_muka LIKE '%" +cari+"%' OR tbl_sisa_pembayaran LIKE
107                 '%" +cari+"%'
108                 + "OR tbl_status_pembayaran LIKE '%" +cari+"%' ORDER BY tbl_id";
109             state=conn.createStatement();
110             res=state.executeQuery(sql);
111             while(res.next()){
112                 tbl_id=res.getString("tbl_id");
113                 tbl_kode_lapangan=res.getString("tbl_kode_lapangan");
114                 tbl_tanggal=res.getString("tbl_tanggal");

```

```

115         tbl_jam=res.getString("tbl_jam");
116         tbl_durasi=res.getString("tbl_durasi");
117         tbl_nama=res.getString("tbl_nama");
118         tbl_telepon=res.getString("tbl_telepon");
119         tbl_uang_muka=res.getString("tbl_uang_muka");
120         tbl_sisa_pembayaran=res.getString("tbl_sisa_pembayaran");
121         tbl_status_pembayaran=res.getString("tbl_status_pembayaran");
122         String [] tampil={" "+tbl_id,tbl_kode_lapangan,tbl_tanggal,tbl_jam,tbl_durasi,tbl_nama
123             ,tbl_telepon,tbl_uang_muka,tbl_sisa_pembayaran,tbl_status_pembayaran};
124         dtm.addRow(tampil);
125     }
126 } catch(SQLException e){
127     e.printStackTrace();
128     JOptionPane.showMessageDialog(null,"Query Salah "+e);
129 }
130 }
131
132 /**
133  * This method is called from within the constructor to initialize the form.
134  * WARNING: Do NOT modify this code. The content of this method is always
135  * regenerated by the Form Editor.
136  */
137 @SuppressWarnings("unchecked")
138 Generated Code
139
308 private void btn_keluarActionPerformed(java.awt.event.ActionEvent evt) {
309     System.exit(0);
310 }
311
312 private void btn_baruActionPerformed(java.awt.event.ActionEvent evt) {
313     baru baru = new baru();
314     baru.setVisible(true);
315     this.dispose();
316 }
317
318 private void btn_perbaruiActionPerformed(java.awt.event.ActionEvent evt) {
319     int i = tbl_data.getSelectedRow();
320     if (i== -1){
321         return;
322     }
323     String code = (String)tbl_data.getValueAt(i,0);
324     String code1 = (String)tbl_data.getValueAt(i,1);
325     String code2 = (String)tbl_data.getValueAt(i,2);
326     String code3 = (String)tbl_data.getValueAt(i,3);
327     String code4 = (String)tbl_data.getValueAt(i,4);
328     String code5 = (String)tbl_data.getValueAt(i,5);
329     String code6 = (String)tbl_data.getValueAt(i,6);
330     String code7 = (String)tbl_data.getValueAt(i,7);

```

331	String code8 = (String)tbl_data.getValueAt(i,8);
332	String code9 = (String)tbl_data.getValueAt(i,9);
333	
334	perbarui pbd = new perbarui();
335	
336	pbd.txt_idsewadb.setText(code);
337	pbd.cmb_kodelapangandb.setSelectedItem(code1);
338	pbd.txt_tanggaldb.setText(code2);
339	pbd.txt_jamdb.setText(code3);
340	pbd.cmb_durasidb.setSelectedItem(code4);
341	pbd.txt_namadb.setText(code5);
342	pbd.txt_nohpdb.setText(code6);
343	pbd.txt_uangmukadb.setText(code7);
344	pbd.txt_sisabayardb.setText(code8);
345	pbd.cmb_statusdb.setSelectedItem(code9);
346	
347	pbd.setVisible(true);
348	this.dispose();
349	}
350	
351	private void tbl_dataMouseClicked(java.awt.event.MouseEvent evt) {
352	
353	}
354	
355	private void btn_hapusActionPerformed(java.awt.event.ActionEvent evt) {
356	hapus();
357	}
358	
359	private void btn_cariActionPerformed(java.awt.event.ActionEvent evt) {
360	cari();
361	}
362	
363	private void btn_bayarActionPerformed(java.awt.event.ActionEvent evt) {
364	int i = tbl_data.getSelectedRow();
365	if (i== -1){
366	return;
367	}
368	
369	String code = (String)tbl_data.getValueAt(i,0);
370	String code1 = (String)tbl_data.getValueAt(i,8);
371	
372	var pls = new pembayaran();
373	
374	pls.txt_idlunas.setText(code);
375	pls.txt_sisabayarbayar.setText(code1);
376	
377	pls.setVisible(true);
378	this.dispose();

379	}
380	
381	/**
382	* @param args the command line arguments
383	*/
384	public static void main(String args[]) {
385	/* Set the Nimbus look and feel */
386	Look and feel setting code (optional)
387	//</editor-fold>
408	
409	/* Create and display the form */
410	java.awt.EventQueue.invokeLater(new Runnable() {
411	public void run() {
412	new data_sewa().setVisible(true);
413	}
414	});
415	}
416	
417	// Variables declaration - do not modify
418	private javax.swing.JButton btn_baru;
419	private javax.swing.JButton btn_bayar;
420	private javax.swing.JButton btn_cari;
421	private javax.swing.JButton btn_hapus;
422	private javax.swing.JButton btn_keluar;
423	private javax.swing.JButton btn_perbarui;
424	private javax.swing.JLabel jLabel1;
425	private javax.swing.JScrollPane jScrollPane1;
426	private javax.swing.JScrollPane jScrollPane2;
427	private javax.swing.JTable jTable1;
428	private javax.swing.JLabel lbl_databooking;
429	private javax.swing.JTable tbl_data;
430	private javax.swing.JTextField txt_cari;
431	// End of variables declaration
432	}

- Kelas baru.java (JFrame Data Baru)

Baris	baru.java
1	import java.awt.HeadlessException;
2	import java.sql.Connection;
3	import java.sql.ResultSet;
4	import java.sql.Statement;
5	import javax.swing.JOptionPane;
6	import java.sql.SQLException;
7	
8	
9	/**
10	*

11	* @author USER
12	*/
13	public class baru extends javax.swing.JFrame {
14	
15	static Connection conn;
16	static ResultSet res;
17	static Statement state;
18	
19	public baru() {
20	initComponents();
21	koneksi DB = new koneksi();
22	DB.config();
23	
24	conn = DB.connection;
25	state = DB.statement;
26	}
27	
28	public void proses_input(){
29	try {
30	state.executeUpdate("INSERT INTO data_sewa (tbl_id, tbl_kode_lapangan"
31	+ ", tbl_tanggal, tbl_jam, tbl_durasi, tbl_nama, tbl_telepon"
32	+ ", tbl_uang_muka, tbl_sisa_pembayaran, tbl_status_pembayaran) VALUES ("
33	+ "" + txt_idsewa.getText()+", "
34	+ "" + cmb_kodelapangan.getSelectedItem()+ ", "
35	+ "" + cmb_tanggal.getSelectedItem()+"-"+cmb_bulan.getSelectedItem()+"-
36	""+cmb_tahun.getSelectedItem()+ ", "
37	+ "" + cmb_waktuawal.getSelectedItem()+"-"+cmb_waktuakhir.getSelectedItem()+ ", "
38	+ "" + cmb_durasi.getSelectedItem()+", "
39	+ "" + txt_nama.getText()+", "
40	+ "" + txt_nohp.getText()+", "
41	+ "" + txt_uangmuka.getText()+", "
42	+ "" + txt_sisabayar.getText()+", "
43	+ "" + cmb_status.getSelectedItem()+ " "));
44	
45	
46	JOptionPane.showMessageDialog(null, "Berhasil Menyimpan Data");
47	data_sewa gui = new data_sewa();
48	gui.setVisible(true);
49	this.dispose();
50	} catch (HeadlessException SQLException e) {
51	JOptionPane.showMessageDialog(null, "Perintah Salah : "+e);
52	}
53	}
54	
55	public void hitung_input(){
56	// Harga Lapangan = Rp. 200.000
57	int a= Integer.parseInt((String)cmb_durasi.getSelectedItem());
58	int b= Integer.parseInt(txt_uangmuka.getText());

```

59     int c= 200000;
60     int d= a*c;
61     int Hasil= d-b;
62     txt_sisabayar.setText(""+Hasil);
63 }
64
65 /**
66  * This method is called from within the constructor to initialize the form.
67  * WARNING: Do NOT modify this code. The content of this method is always
68  * regenerated by the Form Editor.
69  */
70 @SuppressWarnings("unchecked")
335 Generated Code
336
337 private void cmb_waktuakhirActionPerformed(java.awt.event.ActionEvent evt) {
338     // TODO add your handling code here:
339 }
340
341 private void btn_batalActionPerformed(java.awt.event.ActionEvent evt) {
342     data_sewa gui = new data_sewa();
343     gui.setVisible(true);
344     this.dispose();
345 }
346
347 private void txt_idsewaActionPerformed(java.awt.event.ActionEvent evt) {
348     // TODO add your handling code here:
349 }
350
351 private void btn_prosesActionPerformed(java.awt.event.ActionEvent evt) {
352     proses_input();
353 }
354
355 private void txt_uangmukaActionPerformed(java.awt.event.ActionEvent evt) {
356     // TODO add your handling code here:
357 }
358
359 private void cmb_tahunActionPerformed(java.awt.event.ActionEvent evt) {
360     // TODO add your handling code here:
361 }
362
363 private void txt_sisabayarActionPerformed(java.awt.event.ActionEvent evt) {
364
365 }
366
367 private void txt_sisabayarInputMethodTextChanged(java.awt.event.InputMethodEvent evt) {
368
369 }
370

```

371	private void btn_hitungActionPerformed(java.awt.event.ActionEvent evt) {
372	hitung_input();
373	}
374	
375	/**
376	* @param args the command line arguments
377	*/
378	public static void main(String args[]) {
379	/* Set the Nimbus look and feel */
400	Look and feel setting code (optional)
401	
402	/* Create and display the form */
403	java.awt.EventQueue.invokeLater(new Runnable() {
404	public void run() {
405	new baru().setVisible(true);
406	}
407	});
408	}
409	
410	// Variables declaration - do not modify
411	private javax.swing.JButton btn_batal;
412	private javax.swing.JButton btn_hitung;
413	private javax.swing.JButton btn_proses;
414	private javax.swing.JComboBox<String> cmb_bulan;
415	private javax.swing.JComboBox<String> cmb_durasi;
416	private javax.swing.JComboBox<String> cmb_kodelapangan;
417	private javax.swing.JComboBox<String> cmb_status;
418	private javax.swing.JComboBox<String> cmb_tahun;
419	private javax.swing.JComboBox<String> cmb_tanggal;
420	private javax.swing.JComboBox<String> cmb_waktuakhir;
421	private javax.swing.JComboBox<String> cmb_waktuawal;
422	private javax.swing.JLabel lbl_durasi;
423	private javax.swing.JLabel lbl_idsewa;
424	private javax.swing.JLabel lbl_jamsewa;
425	private javax.swing.JLabel lbl_kodelapangan;
426	private javax.swing.JLabel lbl_nama;
427	private javax.swing.JLabel lbl_nohp;
428	private javax.swing.JLabel lbl_perbaruidata;
429	private javax.swing.JLabel lbl_sampai;
430	private javax.swing.JLabel lbl_sisabayar;
431	private javax.swing.JLabel lbl_status;
432	private javax.swing.JLabel lbl_tanggalsewa;
433	private javax.swing.JLabel lbl_uangmuka;
434	private javax.swing.JTextField txt_idsewa;
435	private javax.swing.JTextField txt_nama;
436	private javax.swing.JTextField txt_nohp;
437	private javax.swing.JTextField txt_sisabayar;
438	private javax.swing.JTextField txt_uangmuka;

439	// End of variables declaration }
-----	--------------------------------------

- Kelas perbarui.java (JFrame Pembaruan Data)

Baris	perbarui.java
1	import java.awt.HeadlessException;
2	import java.sql.Connection;
3	import java.sql.ResultSet;
4	import java.sql.Statement;
5	import javax.swing.JOptionPane;
6	import java.sql.SQLException;
7	import java.sql.PreparedStatement;
8	
9	
10	
11	
12	/**
13	*
14	* @author USER
15	*/
16	public class perbarui extends javax.swing.JFrame {
17	
18	static Connection conn;
19	static ResultSet res;
20	static Statement state;
21	
22	public perbarui() {
23	initComponents();
24	koneksi DB = new koneksi();
25	DB.config();
26	
27	conn = DB.connection;
28	state = DB.statement;
29	}
30	
31	public void edit() {
32	int ok=JOptionPane.showConfirmDialog(null,"Apakah Yakin Untuk Update Record
33	ini???", "Confirmation", JOptionPane.YES_NO_OPTION);
34	try {
35	String sql="update data_sewa set
36	tbl_id=?,tbl_kode_lapangan=?,tbl_tanggal=?,tbl_jam=?,tbl_durasi=?,tbl_nama=?"
37	+ ",tbl_telepon=?,tbl_uang_muka=?,tbl_sisa_pembayaran=?,tbl_status_pembayaran=? "
38	+ "where tbl_id='"+txt_idsewadb.getText()+"'";
	PreparedStatement state = conn.prepareStatement(sql);


```

39         if(ok==0) {
40             try {
41                 state.setString(1,txt_idsewadb.getText());
42                 state.setString(2, (String) cmb_kodelapangandb.getSelectedItem());
43                 state.setString(3,txt_tanggaldb.getText());
44                 state.setString(4,txt_jamdb.getText());
45                 state.setString(5,(String) cmb_durasidb.getSelectedItem());
46                 state.setString(6,txt_namadb.getText());
47                 state.setString(7,txt_nohpdb.getText());
48                 state.setString(8,txt_uangmukadb.getText());
49                 state.setString(9,txt_sisabayardb.getText());
50                 state.setString(10, (String) cmb_statusdb.getSelectedItem());
51                 state.executeUpdate();
52
53                 data_sewa ds = new data_sewa();
54                 ds.tampil();
55
56                 JOptionPane.showMessageDialog(null,"Update Data Sukses");
57                 data_sewa gui = new data_sewa();
58                 gui.setVisible(true);
59                 this.dispose();
60             }
61             catch (Exception e) {
62                 JOptionPane.showMessageDialog(null, "Update Data Gagal");
63             }
64         }
65     } catch (Exception e) {}
66 }
67
68 public void hitung_edit(){
69     // Harga Lapangan = Rp. 200.000
70     int a= Integer.parseInt((String)cmb_durasidb.getSelectedItem());
71     int b= Integer.parseInt(txt_uangmukadb.getText());
72     int c= 200000;
73     int d= a*c;
74     int Hasil= d-b;
75     txt_sisabayardb.setText(""+Hasil);
76 }
77
78 /**
79  * This method is called from within the constructor to initialize the form.
80  * WARNING: Do NOT modify this code. The content of this method is always
81  * regenerated by the Form Editor.
82  */
83 @SuppressWarnings("unchecked")
84 Generated Code
85
86 private void btn_batalActionPerformed(java.awt.event.ActionEvent evt) {
306

```

```

307     data_sewa gui = new data_sewa();
308     gui.setVisible(true);
309     this.dispose();
310 }
311
312 private void txt_idsewadbActionPerformed(java.awt.event.ActionEvent evt) {
313     // TODO add your handling code here:
314 }
315
316 private void btn_perbaruiActionPerformed(java.awt.event.ActionEvent evt) {
317     edit();
318 }
319
320 private void txt_uangmukadbActionPerformed(java.awt.event.ActionEvent evt) {
321     // TODO add your handling code here:
322 }
323
324 private void txt_sisabayardbActionPerformed(java.awt.event.ActionEvent evt) {
325
326 }
327
328 private void txt_sisabayardbInputMethodTextChanged(java.awt.event.InputMethodEvent evt) {
329
330 }
331
332 private void btn_hitungdbActionPerformed(java.awt.event.ActionEvent evt) {
333     hitung_edit();
334 }
335
336 /**
337  * @param args the command line arguments
338  */
339 public static void main(String args[]) {
340     /* Set the Nimbus look and feel */
341     Look and feel setting code (optional)
342
343     /* Create and display the form */
344     java.awt.EventQueue.invokeLater(new Runnable() {
345         public void run() {
346             new baru().setVisible(true);
347         }
348     });
349 }
350
351 // Variables declaration - do not modify
352 private javax.swing.JButton btn_batal;
353 private javax.swing.JButton btn_hitungdb;
354 private javax.swing.JButton btn_perbarui;

```

375	public javax.swing.JComboBox<String> cmb_durasidb;
376	public javax.swing.JComboBox<String> cmb_kodelapangandb;
377	public javax.swing.JComboBox<String> cmb_statusdb;
378	private javax.swing.JLabel lbl_durasi;
379	private javax.swing.JLabel lbl_idsewadb;
380	private javax.swing.JLabel lbl_jamsewa;
381	private javax.swing.JLabel lbl_kodelapangandb;
382	private javax.swing.JLabel lbl_nama;
383	private javax.swing.JLabel lbl_nohp;
384	private javax.swing.JLabel lbl_perbaruidata;
385	private javax.swing.JLabel lbl_sisabayar;
386	private javax.swing.JLabel lbl_status;
387	private javax.swing.JLabel lbl_tanggalsewadb;
388	private javax.swing.JLabel lbl_uangmuka;
390	public javax.swing.JTextField txt_idsewadb;
391	public javax.swing.JTextField txt_jamdb;
392	public javax.swing.JTextField txt_namadb;
393	public javax.swing.JTextField txt_nohpdb;
394	public javax.swing.JTextField txt_sisabayardb;
395	public javax.swing.JTextField txt_tanggaldb;
396	public javax.swing.JTextField txt_uangmukadb;
397	// End of variables declaration
398	}
399	

- Kelas pembayaran.java (JFrame Pembayaran)

Baris	login.java
1	import java.awt.HeadlessException;
2	import java.sql.Connection;
3	import java.sql.ResultSet;
4	import java.sql.Statement;
5	import javax.swing.JOptionPane;
6	import java.sql.SQLException;
7	import java.sql.PreparedStatement;
8	
9	
10	
11	
12	/**
13	*
14	* @author USER
15	*/
16	public class pembayaran extends javax.swing.JFrame {
17	
18	static Connection conn;
19	static ResultSet res;
20	static Statement state;
21	
22	public pembayaran() {
23	initComponents();
24	koneksi DB = new koneksi();
25	DB.config();
26	
27	conn = DB.connection;
28	state = DB.statement;
29	}
30	
31	public void bayar() {
32	int ok=JOptionPane.showConfirmDialog(null,"Apakah Yakin Booking sebelumnya ingin di Bayar
33	???", "Confirmation", JOptionPane.YES_NO_OPTION);
34	try {
35	String sql="UPDATE data_sewa SET tbl_id=?, tbl_uang_muka=?, tbl_sisa_pembayaran=?,
36	tbl_status_pembayaran=?"
37	+ "WHERE tbl_id='"+txt_idlunas.getText()+"'";
38	PreparedStatement state = conn.prepareStatement(sql);
39	if(ok==0) {
40	try {
41	state.setString(1,txt_idlunas.getText());
42	state.setString(2,txt_uangmasukbayar.getText());
43	state.setString(3,txt_sisabayarbayar.getText());
44	state.setString(4, (String) cmb_statusbayar.getSelectedItem());
45	state.executeUpdate();

```

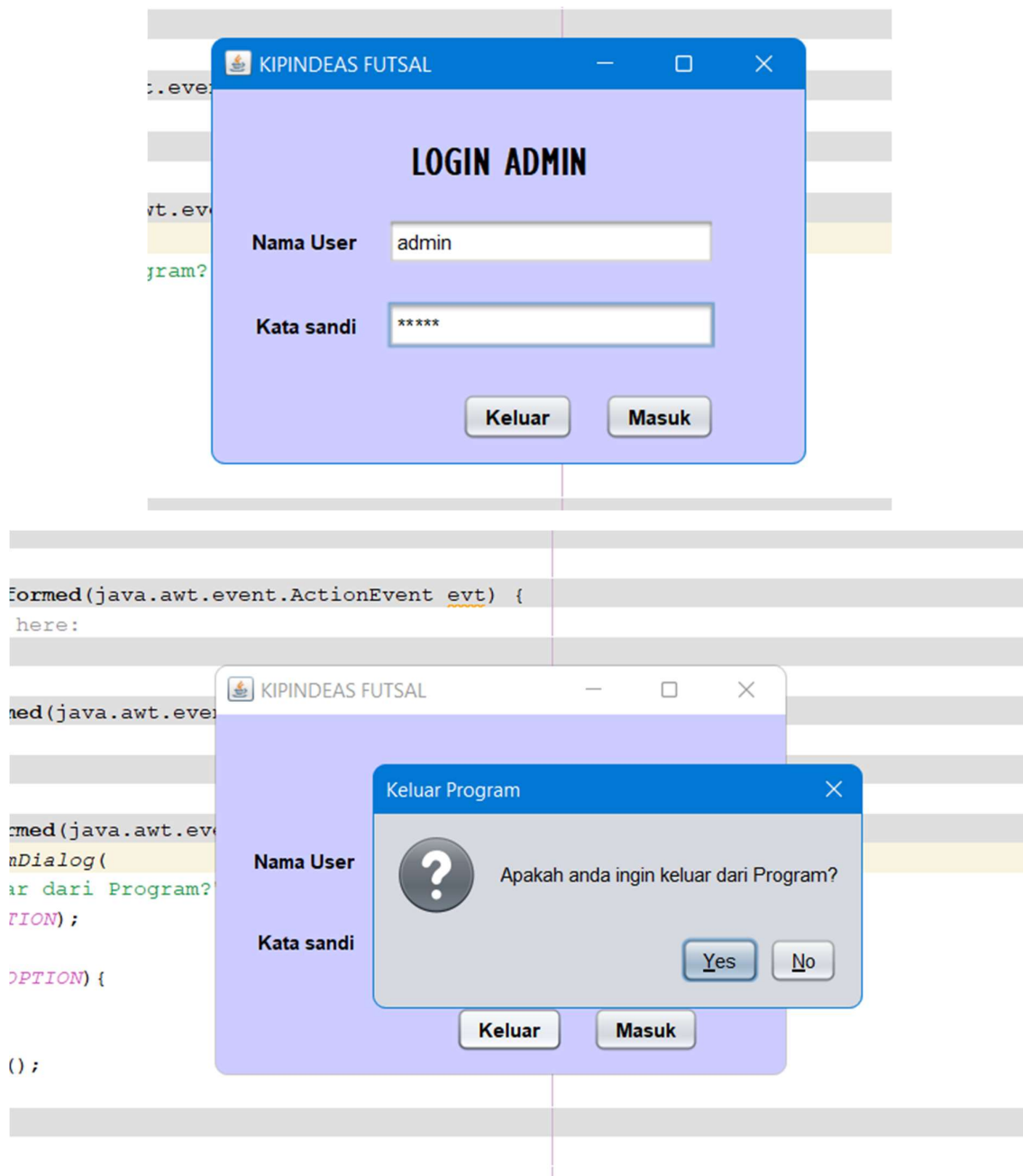
46         data_sewa ds = new data_sewa();
47         ds.tampil();
48
49         JOptionPane.showMessageDialog(null,"Update Data Sukses");
50         data_sewa gui = new data_sewa();
51         gui.setVisible(true);
52         this.dispose();
53     }
54     catch (Exception e) {
55         JOptionPane.showMessageDialog(null, "Update Data Gagal");
56     }
57 }
58 } catch (Exception e){}
59 }
60
61 public void hitung_bayar(){
62     int a= Integer.parseInt(txt_uangmasukbayar.getText());
63     int b= Integer.parseInt(txt_sisabayarbayar.getText());
64     int Hasil= b-a;
65     txt_sisabayarbayar.setText(""+Hasil);
66 }
67
68 /**
69  * This method is called from within the constructor to initialize the form.
70  * WARNING: Do NOT modify this code. The content of this method is always
71  * regenerated by the Form Editor.
72  */
73 @SuppressWarnings("unchecked")
74
226 private void btn_batalActionPerformed(java.awt.event.ActionEvent evt) {
227     data_sewa gui = new data_sewa();
228     gui.setVisible(true);
229     this.dispose();
230 }
231
232 private void btn_bayarActionPerformed(java.awt.event.ActionEvent evt) {
233     bayar();
234 }
235
236 private void txt_sisabayarbayarActionPerformed(java.awt.event.ActionEvent evt) {
237
238 }
239
240 private void txt_sisabayarbayarInputMethodTextChanged(java.awt.event.InputMethodEvent evt) {
241
242 }
243
244 private void btn_hitungbayarActionPerformed(java.awt.event.ActionEvent evt) {

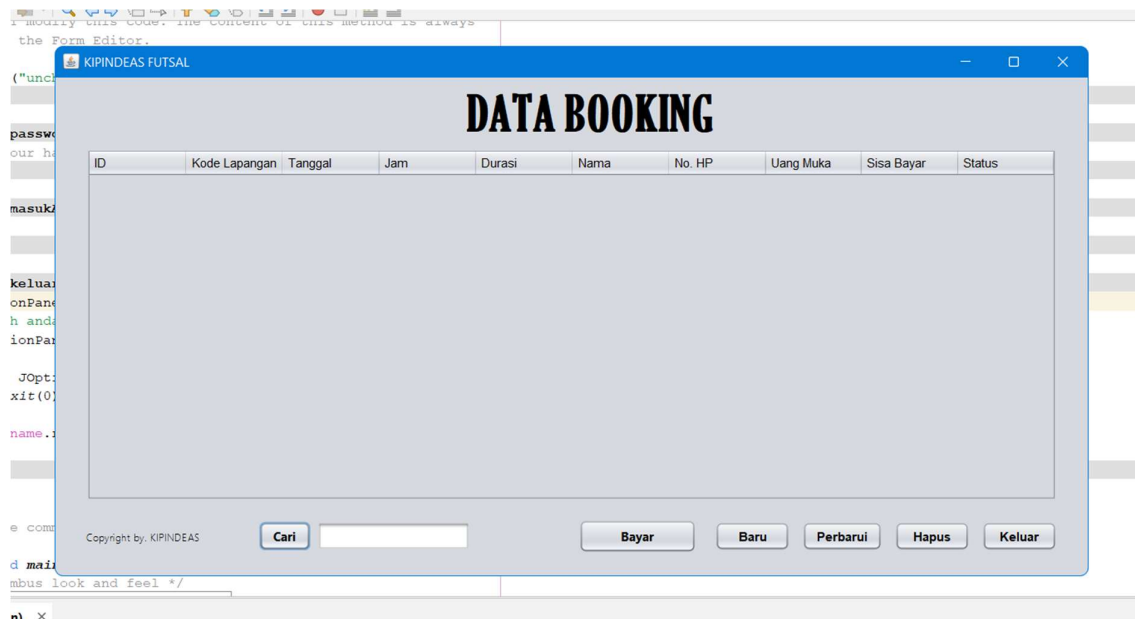
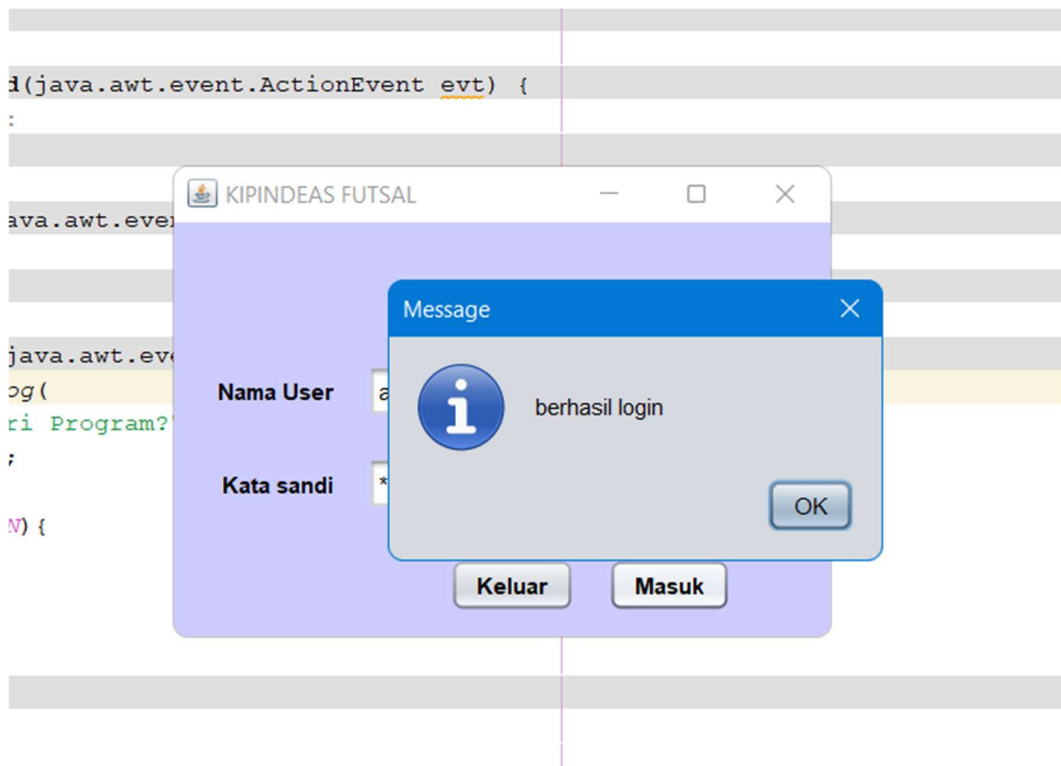
```

245	hitung_bayar();
246	}
247	
248	private void txt_uangmasukbayarActionPerformed(java.awt.event.ActionEvent evt) {
249	// TODO add your handling code here:
250	}
251	
252	/**
253	* @param args the command line arguments
254	*/
255	public static void main(String args[]) {
256	/* Set the Nimbus look and feel */
257	Look and feel setting code (optional)
258	
259	/* Create and display the form */
260	java.awt.EventQueue.invokeLater(new Runnable() {
261	public void run() {
262	new baru().setVisible(true);
263	}
264	});
265	}
266	
267	// Variables declaration - do not modify
268	private javax.swing.JButton btn_batal;
269	private javax.swing.JButton btn_bayar;
270	private javax.swing.JButton btn_hitungbayar;
271	public javax.swing.JComboBox<String> cmb_statusbayar;
272	private javax.swing.JLabel lbl_perbaruidata;
273	private javax.swing.JLabel lbl_sisabayar;
274	private javax.swing.JLabel lbl_status;
275	private javax.swing.JLabel lbl_uangmasuk;
276	private javax.swing.JLabel lbl_uangmuka;
277	public javax.swing.JTextField txt_idlunas;
278	public javax.swing.JTextField txt_sisabayarbayar;
279	public javax.swing.JTextField txt_uangmasukbayar;
280	// End of variables declaration
281	}
282	

2.6 Screenshot

Berikut merupakan screenshot dari aplikasi yang berjalan, dari screenshot ini sudah diurutkan dari awal hingga akhir.





KIPINDEAS FUTSAL

DATA BARU

ID Sewa: 1

Kode Lapangan: LAP001

Tanggal Sewa: 03 Maret 2022

Jam Sewa: 15:00 Sampai 17:00

Durasi (Jam): 2

Nama: Azki

No HP: 087786469213

Uang Muka: 200000

Sisa Bayar: 200000

Status: Belum Lunas

KIPINDEAS FUTSAL

DATA BARU

ID Sewa: 1

Kode Lapangan: LAP001

Tanggal Sewa: 03 Maret 2022

Jam Sewa: 15:00 Sampai 17:00

Durasi (Jam): 2

Nama: Azki


No HP: 087786469213

Uang Muka: 200000

Sisa Bayar: 200000

Status: Belum Lunas

Message

 Berhasil Menyimpan Data

KIPINDEAS FUTSAL

DATA BOOKING

ID	Kode Lapangan	Tanggal	Jam	Durasi	Nama	No. HP	Uang Muka	Sisa Bayar	Status
1	LAP001	03-Maret-2022	15:00-17:00	2	Azki	087786469213	200000	200000	Belum Lunas

Copyright by: KIPINDEAS

KIPINDEAS FUTSAL

DIPERBARUI DATA

ID Sewa:

Kode Lapangan:

Tanggal Sewa:

Jam Sewa:

Durasi (Jam):

Nama:

No HP:

Uang Muka:

Sisa Bayar:

Status:

KIPINDEAS FUTSAL

DIPERBARUI DATA


ID Sewa	<input type="text" value="1"/>
Kode Lapangan	<input type="text" value="LAP003"/>
Tanggal Sewa	<input type="text" value="03-Maret-2022"/>
Jam Sewa	<input type="text" value="15:00-17:00"/>
Durasi (Jam)	<input type="text" value="2"/>
Nama	<input type="text" value="Azki Nurul Fajri"/>
No HP	<input type="text" value="087786469925"/>
Uang Muka	<input type="text" value="200000"/>
Sisa Bayar	<input type="text" value="200000"/> <input type="button" value="Hitung"/>
Status	<input type="text" value="Belum Lunas"/>

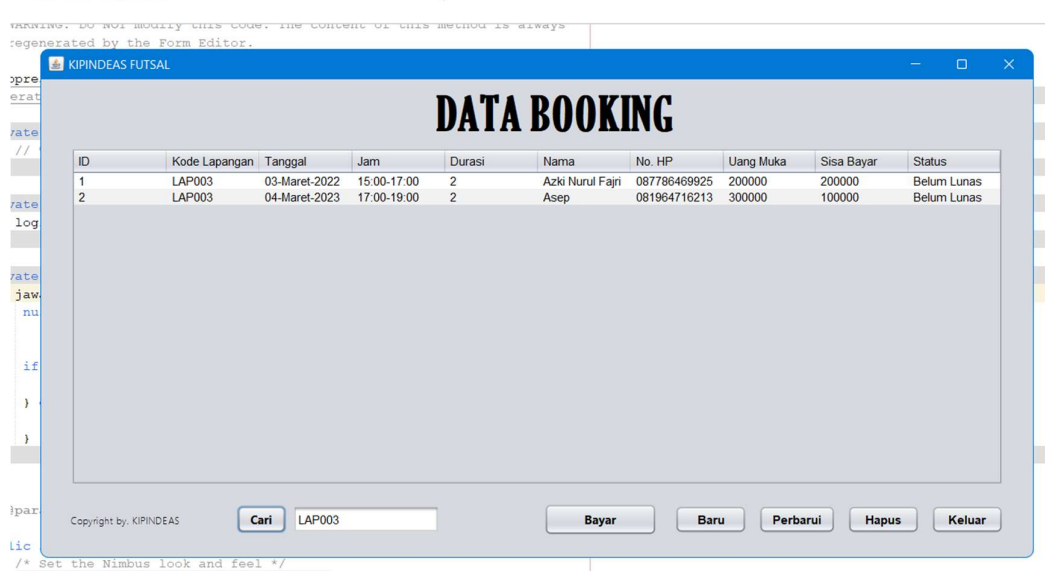
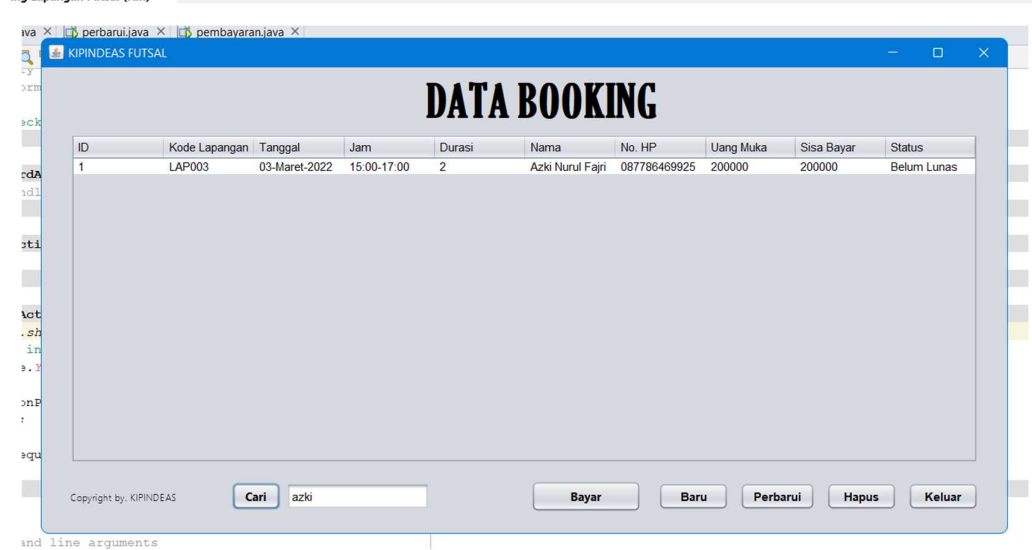
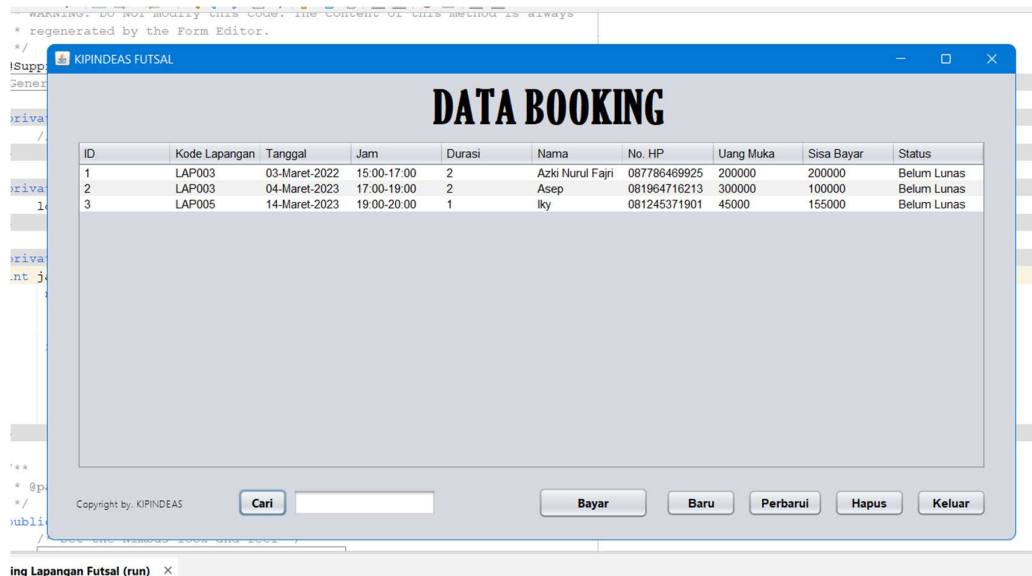
KIPINDEAS FUTSAL

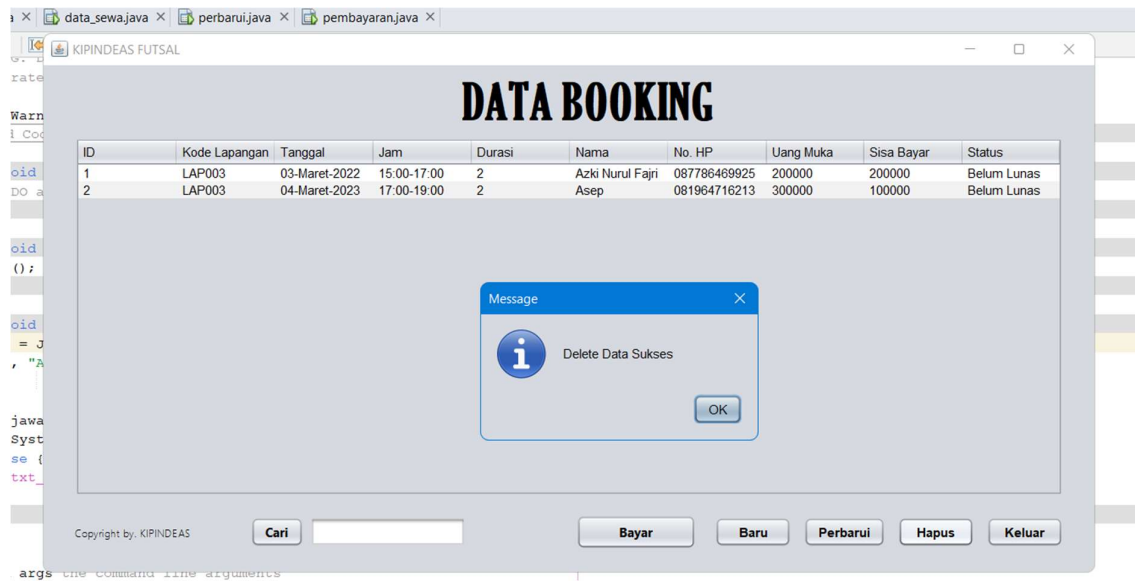
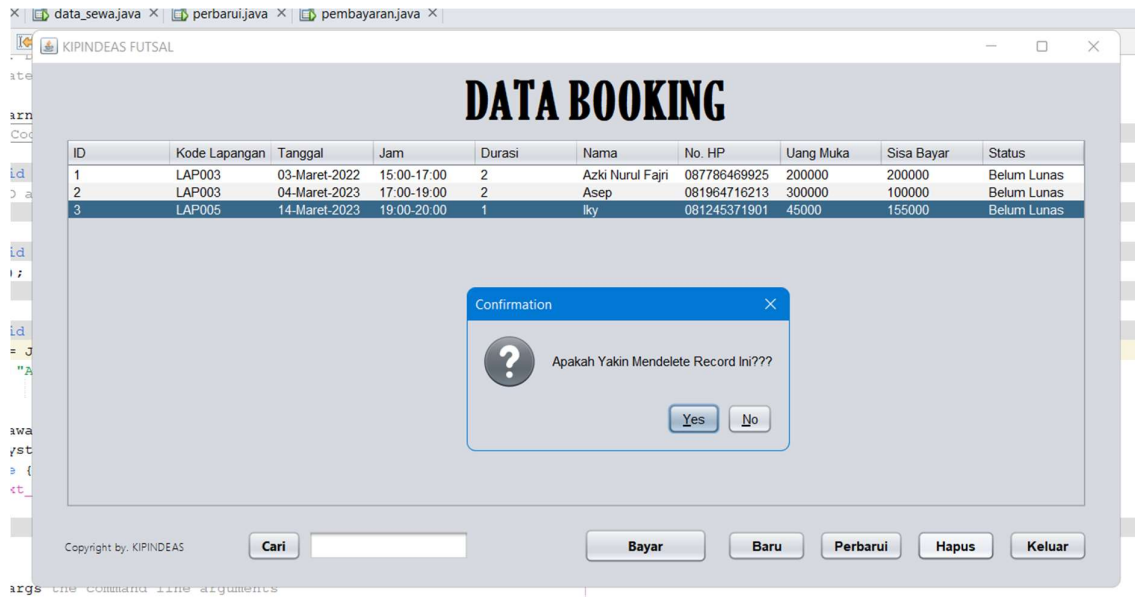
DIPERBARUI DATA

ID Sewa	<input type="text" value="1"/>
Kode Lapangan	<input type="text" value="LAP003"/>
Tanggal Sewa	<input type="text" value="03-Maret-2022"/>
Jam Sewa	<input type="text" value="15:00-17:00"/>
Durasi (Jam)	<input type="text" value="2"/>
Nama	<input type="text" value="Azki Nurul Fajri"/>
No HP	<input type="text" value="087786469925"/>
Uang Muka	<input type="text" value="200000"/>
Sisa Bayar	<input type="text" value="200000"/> <input type="button" value="Hitung"/>
Status	<input type="text" value="Belum Lunas"/>

Message

 Update Data Sukses





KIPINDEAS FUTSAL

PEMBAYARAN

ID

Uang Masuk

Sisa Bayar

Status

KIPINDEAS FUTSAL

PEMBAYARAN

ID

Uang Masuk

Sisa Bayar

Status

KIPINDEAS FUTSAL

DATA BOOKING

ID	Kode Lapangan	Tanggal	Jam	Durasi	Nama	No. HP	Uang Muka	Sisa Bayar	Status
1	LAP003	03-Maret-2022	15:00-17:00	2	Azki Nurul Fajri	087786469925	200000	200000	Belum Lunas
2	LAP003	04-Maret-2023	17:00-19:00	2	Asep	081964716213	100000	0	Sudah Lunas

Copyright by: KIPINDEAS

Cari

Bayar

Baru

Perbarui

Hapus

Keluar