2.4 Design GUI

2.5 Penjelasan Fitur-fitur pada GUI

# 2.6 Kode Program

|  |  |
| --- | --- |
| Class DatabaseConfig | |
|  | package ConfigDatabase;  import java.sql.\*;  public class DatabaseConfig {  private Connection connection;  public Connection getConnection() {  String jdbcUrl = "jdbc:mysql://localhost/wanmarized";  String username = "root";  String password = "";  try {  connection = DriverManager.getConnection(jdbcUrl, username, password);  } catch (SQLException e) {  e.printStackTrace();  }  return connection;  }  } |

|  |  |
| --- | --- |
| Class LoginPage | |
|  | package wanmarized;  import java.sql.\*;  import ConfigDatabase.DatabaseConfig;  import javax.swing.JOptionPane;  public class LoginPage extends javax.swing.JFrame {  public LoginPage() {  initComponents();  setLocationRelativeTo(null);  }  @SuppressWarnings("unchecked")  // Generated Code  private void login() {  String username = kolomid.getText();  String password = new String(kolompassword.getPassword());  String query = "SELECT username,password FROM user WHERE username= ? AND password=?";  try (Connection connection = new DatabaseConfig().getConnection(); PreparedStatement pst = connection.prepareStatement(query)) {  pst.setString(1, username);  pst.setString(2, password);  //User user = new User();  try (ResultSet resultSet = pst.executeQuery()) {  if (resultSet.next()) {  Dashboard dashboard = new Dashboard();  setVisible(false);  dispose();  dashboard.setVisible(true);  } else {  JOptionPane.showMessageDialog(null, "ID atau Password yang anda masukkan Salah!");  }  }  } catch (SQLException ex) {  JOptionPane.showMessageDialog(null, ex.getMessage());  }  }  private void loginbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  this.login();  }  private void exitbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  int reply = JOptionPane.showConfirmDialog(null, "Yakin ingin keluar?", "Konfirmasi Keluar Aplikasi", JOptionPane.YES\_NO\_OPTION);  if (reply == JOptionPane.YES\_OPTION) {  System.exit(0);  } else {  kolomid.requestFocus();  }  }  private void signupbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  SignupPage signup = new SignupPage();  setVisible(false);  dispose();  signup.setVisible(true);  }  private void kolomidActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  this.login();  }  private void kolompasswordActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  this.login();  }  public static void main(String args[]) {  /\* Create and display the form \*/  java.awt.EventQueue.invokeLater(new Runnable() {  public void run() {  new LoginPage().setVisible(true);  }  });  }  // Variables declaration - do not modify  private javax.swing.JButton exitbtn;  private javax.swing.JLabel idLabel;  private javax.swing.JPanel jPanel1;  private javax.swing.JPanel jPanel2;  private javax.swing.JPanel jPanel3;  private javax.swing.JPanel jPanel4;  private javax.swing.JPanel jPanel5;  private javax.swing.JPanel jPanel6;  private javax.swing.JPanel jPanel7;  private javax.swing.JPanel jPanel8;  private javax.swing.JTextField kolomid;  private javax.swing.JPasswordField kolompassword;  private javax.swing.JButton loginbtn;  private javax.swing.JLabel pageTitle;  private javax.swing.JLabel passwordLabel;  private javax.swing.JButton signupbtn;  // End of variables declaration  } |

|  |  |
| --- | --- |
| Class SignupPage | |
|  | package wanmarized;  import java.sql.\*;  import java.util.logging.Level;  import java.util.logging.Logger;  import javax.swing.JOptionPane;  import ConfigDatabase.DatabaseConfig;  public class SignupPage extends javax.swing.JFrame {  Connection con;  PreparedStatement pst1;  public SignupPage() {  initComponents();  setLocationRelativeTo(null);  }  @SuppressWarnings("unchecked")  // Generated Code  private void kembaliKeMenuLogin() {  LoginPage login = new LoginPage();  setVisible(false);  dispose();  login.setVisible(true);  }  private void daftarAkun() {  if (kolomid.getText().equals("") || new String(kolompassword.getPassword()).equals("")) {  JOptionPane.showMessageDialog(this, "Isi kolom id & password terlebih dahulu!");  return;  }  int reply = JOptionPane.showConfirmDialog(null, "Apakah data yang anda masukkan sudah benar?", "Konfirmasi Pendaftaran Akun", JOptionPane.YES\_NO\_OPTION);  if (reply == JOptionPane.YES\_OPTION) {  try {  tambahkanDataAkunKeDatabase();  } catch (ClassNotFoundException ex) {  Logger.getLogger(Dashboard.class.getName()).log(Level.SEVERE, null, ex);  }  }  }  private void backbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  this.kembaliKeMenuLogin();  }  private void signbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  this.daftarAkun();  }  private void tambahkanDataAkunKeDatabase() throws ClassNotFoundException {  String id = kolomid.getText();  String password = new String(kolompassword.getPassword());  String query = "insert into user values(?,?)";  try (Connection connection = new DatabaseConfig().getConnection(); PreparedStatement pst = connection.prepareStatement(query)) {  pst.setString(1, id);  pst.setString(2, password);  pst.executeUpdate();  JOptionPane.showMessageDialog(this, "Pendaftaran anda berhasil!");  this.clearInput();  } catch (SQLException ex) {  Logger.getLogger(Dashboard.class.getName()).log(Level.SEVERE, null, ex);  }  }  private void clearInput() {  kolomid.setText("");  kolompassword.setText("");  }  public static void main(String args[]) {  java.awt.EventQueue.invokeLater(new Runnable() {  public void run() {  SignupPage signup = new SignupPage();  signup.setVisible(true);  }  });  }  // Variables declaration - do not modify  private javax.swing.JButton backbtn;  private javax.swing.JLabel jLabel1;  private javax.swing.JLabel jLabel2;  private javax.swing.JLabel jLabel3;  private javax.swing.JPanel jPanel1;  private javax.swing.JPanel jPanel2;  private javax.swing.JTextField kolomid;  private javax.swing.JPasswordField kolompassword;  private javax.swing.JButton signbtn;  // End of variables declaration  } |

|  |  |
| --- | --- |
| Class Dashboard | |
|  | package wanmarized;  import java.awt.Component;  import java.awt.Container;  import java.awt.Dimension;  import java.awt.FlowLayout;  import java.io.BufferedWriter;  import java.io.FileWriter;  import java.sql.Statement;  import java.sql.Connection;  import java.util.logging.Level;  import java.util.logging.Logger;  import java.sql.DriverManager;  import java.sql.SQLException;  import java.sql.PreparedStatement;  import java.sql.ResultSet;  import java.sql.ResultSetMetaData;  import java.util.ArrayList;  import java.util.Vector;  import javax.swing.\*;  import javax.swing.table.DefaultTableModel;  import ConfigDatabase.DatabaseConfig;  import java.util.HashMap;  import net.sf.jasperreports.engine.JRException;  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;  public class Dashboard extends javax.swing.JFrame {  Connection con;  PreparedStatement pst1;  PreparedStatement pst2;  DefaultTableModel model = new DefaultTableModel();  ArrayList<javax.swing.JCheckBox> foodCheckbox = new ArrayList<javax.swing.JCheckBox>();  ArrayList<javax.swing.JLabel> foodLabel = new ArrayList<javax.swing.JLabel>();  ArrayList<javax.swing.JSpinner> foodQty = new ArrayList<javax.swing.JSpinner>();  public Dashboard() {  initComponents();  this.mengaturLayoutForm();  }  @SuppressWarnings("unchecked")  // Generated Code  private void resetDashboard() {  while (model.getRowCount() > 0) {  model.removeRow(0);  }  txtsub.setText("");  txtpayment.setText("");  txtreturn.setText("");  for (int i = 0; i < foodCheckbox.size(); i++) {  foodCheckbox.get(i).setSelected(false);  foodQty.get(i).setValue(0);  }  }  private void mengaturLayoutForm() {  String query = "SELECT nama\_produk, id\_produk FROM produk";  try (Connection connection = new DatabaseConfig().getConnection(); PreparedStatement pst = connection.prepareStatement(query); ResultSet resultSet = pst.executeQuery()) {  form.setLayout(new BoxLayout(form, BoxLayout.Y\_AXIS)); // Ganti layout manager menjadi Y\_AXIS  while (resultSet.next()) {  String idProduk = resultSet.getString("id\_produk");  String namaProduk = resultSet.getString("nama\_produk");  // Buat dan konfigurasi JPanel untuk grup komponen  javax.swing.JPanel group = new javax.swing.JPanel();  group.setLayout(new LeftFlowLayout(60)); // Ganti layout manager menjadi FlowLayout  // Buat dan konfigurasi JCheckBox  javax.swing.JCheckBox foodName = new javax.swing.JCheckBox();  foodName.setFont(new java.awt.Font("Helvetica Neue", 1, 12));  foodName.setText(idProduk);  foodName.setPreferredSize(new java.awt.Dimension(150, 30));  javax.swing.JLabel foodNameLable = new javax.swing.JLabel();  foodNameLable.setFont(new java.awt.Font("Helvetica Neue", 0, 14)); // NOI18N  foodNameLable.setText(namaProduk);  foodNameLable.setPreferredSize(new java.awt.Dimension(100, 30));  // Buat dan konfigurasi JSpinner  javax.swing.JSpinner qty = new javax.swing.JSpinner();  qty.setPreferredSize(new java.awt.Dimension(50, 30));  // Tambahkan komponen-komponen ke grup  group.add(foodName);  group.add(foodNameLable);  group.add(qty);  // Menambahkan component ke Array List  foodCheckbox.add(foodName);  foodLabel.add(foodNameLable);  foodQty.add(qty);  // Tambahkan grup ke panel utama dengan menggunakan GridBagConstraints  form.add(group);  }  } catch (SQLException ex) {  JOptionPane.showMessageDialog(null, ex.getMessage());  }  }  private void logoutbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  int reply = JOptionPane.showConfirmDialog(this, "Apakah anda yakin ingin keluar?", "Logout", JOptionPane.YES\_NO\_OPTION);  if (reply == JOptionPane.YES\_OPTION) {  LoginPage login = new LoginPage();  setVisible(false);  dispose();  login.setVisible(true);  } else {  }  }  private void addTransactionBtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  int price = 0;  int sum = 0;  model = (DefaultTableModel) transactionTable.getModel();  for (int i = 0; i < foodCheckbox.size(); i++) {  if (foodCheckbox.get(i).isSelected()) {  if (foodQty.get(i).getValue().toString().equals("0")) {  JOptionPane.showMessageDialog(this, "Masih ada makanan yang belum diisi kuantitasnya.");  return;  };  }  }  for (int i = 0; i < foodCheckbox.size(); i++) {  if (foodCheckbox.get(i).isSelected()) {  String query1 = "Select harga\_jual from produk where id\_produk= ?";  try (Connection connection = new DatabaseConfig().getConnection(); PreparedStatement pst = connection.prepareStatement(query1)) {  pst.setString(1, foodCheckbox.get(i).getText());  try (ResultSet resultSet = pst.executeQuery()) {  if (resultSet.next()) {  price = resultSet.getInt("harga\_jual");  }  }  } catch (SQLException e) {  JOptionPane.showMessageDialog(null, e.getMessage());  }  int qty = Integer.parseInt(foodQty.get(i).getValue().toString());  int total = price \* qty;  model = (DefaultTableModel) transactionTable.getModel();  model.addRow(new Object[]{  foodCheckbox.get(i).getText(),  price,  qty,  total  });  }  }  for (int i = 0; i < transactionTable.getRowCount(); i++) {  sum += Integer.parseInt(transactionTable.getValueAt(i, 3).toString());  }  txtsub.setText(Integer.toString(sum));  }  private void printInvoiceBtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  int sub, pay, bal;  sub = Integer.parseInt(txtsub.getText());  pay = Integer.parseInt(txtpayment.getText());  bal = pay - sub;  txtreturn.setText(String.valueOf(bal));  try {  simpanDataTransaksi();  this.resetDashboard();  } catch (ClassNotFoundException ex) {  Logger.getLogger(Dashboard.class.getName()).log(Level.SEVERE, null, ex);  } catch (JRException ex) {  Logger.getLogger(Dashboard.class.getName()).log(Level.SEVERE, null, ex);  }  }  private void foodMenuBtnActionPerformed(java.awt.event.ActionEvent evt) {  MenuMakanan menumakanan = new MenuMakanan();  setVisible(false);  dispose();  menumakanan.setVisible(true);  }  class LeftFlowLayout extends FlowLayout {  private static final long serialVersionUID = 1L;  private int marginRight;  public LeftFlowLayout(int marginRight) {  super(FlowLayout.LEFT);  this.marginRight = marginRight;  }  @Override  public void layoutContainer(Container target) {  synchronized (target.getTreeLock()) {  int n = target.getComponentCount();  if (n > 0) {  Component c = target.getComponent(0);  Dimension preferredSize = c.getPreferredSize();  int x = target.getInsets().left;  int y = target.getInsets().top + (target.getHeight() - target.getInsets().top - target.getInsets().bottom - preferredSize.height) / 2;  c.setBounds(x, y, preferredSize.width, preferredSize.height);  for (int i = 1; i < n; i++) {  Component prev = target.getComponent(i - 1);  c = target.getComponent(i);  preferredSize = c.getPreferredSize();  x = prev.getX() + prev.getWidth() + marginRight;  y = target.getInsets().top + (target.getHeight() - target.getInsets().top - target.getInsets().bottom - preferredSize.height) / 2;  c.setBounds(x, y, preferredSize.width, preferredSize.height);  }  }  }  }  }  public void simpanDataTransaksi() throws ClassNotFoundException, JRException {  String subtot = txtsub.getText();  String pay = txtpayment.getText();  String bal = txtreturn.getText();  int lastinsertid = 0;  try {  Class.forName("com.mysql.jdbc.Driver");  con = DriverManager.getConnection("jdbc:mysql://localhost/wanmarized", "root", "");  String query2 = "insert into catatan\_penjualan(subtotal,pay,balance)values(?,?,?)";  pst1 = con.prepareStatement(query2, Statement.RETURN\_GENERATED\_KEYS);  pst1.setString(1, subtot);  pst1.setString(2, pay);  pst1.setString(3, bal);  pst1.executeUpdate();  ResultSet generatekey = pst1.getGeneratedKeys();  if (generatekey.next()) {  lastinsertid = generatekey.getInt(1);  }  String query1 = "insert into produk\_terjual(sales\_id,id\_produk,price,qty,total)values(?,?,?,?,?)";  pst2 = con.prepareStatement(query1);  String product = "";  int price = 0;  int qty = 0;  int tot = 0;  for (int i = 0; i < transactionTable.getRowCount(); i++) {  product = (String) transactionTable.getValueAt(i, 0);  price = (int) transactionTable.getValueAt(i, 1);  qty = (int) transactionTable.getValueAt(i, 2);  tot = (int) transactionTable.getValueAt(i, 3);  pst2.setInt(1, lastinsertid);  pst2.setString(2, product);  pst2.setInt(3, price);  pst2.setInt(4, qty);  pst2.setInt(5, tot);  pst2.executeUpdate();  }  JOptionPane.showMessageDialog(this, "Transaksi selesai");  HashMap a = new HashMap();  a.put("invo", lastinsertid);  try {  JasperDesign jdesign = JRXmlLoader.load("/Users/macbookair/JaspersoftWorkspace/StructReport/Wanmarized\_Struct.jrxml");  JasperReport jreport = JasperCompileManager.compileReport(jdesign);  System.out.println(jreport);  JasperPrint jprint = JasperFillManager.fillReport(jreport, a, con);  JasperViewer view = new JasperViewer(jprint, false);  view.setVisible(true);  } catch (JRException ex) {  Logger.getLogger(Dashboard.class.getName()).log(Level.SEVERE, null, ex);  }  } catch (SQLException ex) {  Logger.getLogger(Dashboard.class.getName()).log(Level.SEVERE, null, ex);  }  }  public static void main(String args[]) {  java.awt.EventQueue.invokeLater(new Runnable() {  @Override  public void run() {  Dashboard dashboard = new Dashboard();  dashboard.setVisible(true);  }  });  }  // Variables declaration - do not modify  private javax.swing.JButton addTransactionBtn;  private javax.swing.JButton foodMenuBtn;  private javax.swing.JPanel form;  private javax.swing.JLabel jLabel1;  private javax.swing.JLabel jLabel2;  private javax.swing.JLabel jLabel3;  private javax.swing.JLabel jLabel4;  private javax.swing.JMenuItem jMenuItem1;  private javax.swing.JPanel jPanel1;  private javax.swing.JPanel jPanel2;  private javax.swing.JPanel jPanel3;  private javax.swing.JScrollPane jScrollPane1;  private javax.swing.JScrollPane jScrollPane2;  private javax.swing.JButton logoutbtn;  private javax.swing.JButton printInvoiceBtn;  private javax.swing.JTable transactionTable;  private javax.swing.JTextField txtpayment;  private javax.swing.JTextField txtreturn;  private javax.swing.JTextField txtsub;  // End of variables declaration  } |

|  |  |
| --- | --- |
| Class MenuMakanan | |
|  | package wanmarized;  import java.sql.\*;  import java.util.ArrayList;  import java.util.Vector;  import java.util.logging.Level;  import java.util.logging.Logger;  import javax.swing.JOptionPane;  import javax.swing.table.DefaultTableModel;  import ConfigDatabase.DatabaseConfig;  public class MenuMakanan extends javax.swing.JFrame {  static public MenuMakanan menuMakanan;  public MenuMakanan() {  initComponents();  setLocationRelativeTo(null);  this.table\_update();  }  @SuppressWarnings("unchecked")  // Generated Code  private void deletebtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  int selectedIndex = jTable1.getSelectedRow();  if (selectedIndex == -1) {  JOptionPane.showMessageDialog(this, "Mohon pilih produk terlebih dahulu.");  return;  }  int dialogResult = JOptionPane.showConfirmDialog(null, "Apakah kamu yakin ingin menghapus produk ini?", "Warning", JOptionPane.YES\_NO\_OPTION);  if (dialogResult == JOptionPane.NO\_OPTION) {  return;  }  DefaultTableModel model = (DefaultTableModel) jTable1.getModel();  String id = model.getValueAt(selectedIndex, 0).toString();  String query = "delete from produk where id\_produk = ?";  try (Connection connection = new DatabaseConfig().getConnection(); PreparedStatement pst = connection.prepareStatement(query)) {  pst.setString(1, id);  pst.executeUpdate();  JOptionPane.showMessageDialog(this, "Data produk berhasil dihapus.");  this.table\_update();  } catch (SQLException ex) {  Logger.getLogger(TambahMakanan.class.getName()).log(Level.SEVERE, null, ex);  }  }  private void addbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  TambahMakanan tambahmakanan = new TambahMakanan();  tambahmakanan.setVisible(true);  }  private void backbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  Dashboard dashboard = new Dashboard();  setVisible(false);  dispose();  dashboard.setVisible(true);  }  private void editbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  int selectedIndex = jTable1.getSelectedRow();  if (selectedIndex == -1) {  JOptionPane.showMessageDialog(this, "Mohon pilih produk terlebih dahulu.");  return;  }    String selectedProductId = jTable1.getValueAt(selectedIndex, 0).toString();  String selectedProductName = jTable1.getValueAt(selectedIndex, 1).toString();  String selectedProductSellingPrice = jTable1.getValueAt(selectedIndex, 2).toString();  String selectedProductCapitalPrice = jTable1.getValueAt(selectedIndex, 3).toString();  EditMakanan editmenu = new EditMakanan(selectedProductId, selectedProductName, selectedProductSellingPrice, selectedProductCapitalPrice);  editmenu.setVisible(true);  }  private void table\_update() {  int jumlahKolom;  String query = "SELECT \* from produk";  try (Connection connection = new DatabaseConfig().getConnection(); PreparedStatement pst = connection.prepareStatement(query); ResultSet resultSet = pst.executeQuery()) {  ResultSetMetaData RSMD = resultSet.getMetaData();  jumlahKolom = RSMD.getColumnCount();  DefaultTableModel DFT = (DefaultTableModel) jTable1.getModel();  DFT.setRowCount(0);  while (resultSet.next()) {  Vector data = new Vector();  for (int i = 1; i <= jumlahKolom; i++) {  data.add(resultSet.getString("id\_produk"));  data.add(resultSet.getString("nama\_produk"));  data.add(resultSet.getString("harga\_jual"));  data.add(resultSet.getString("harga\_modal"));  }  DFT.addRow(data);  }  } catch (SQLException ex) {  Logger.getLogger(MenuMakanan.class.getName()).log(Level.SEVERE, null, ex);  }  }  public static void main(String args[]) {  java.awt.EventQueue.invokeLater(new Runnable() {  public void run() {  menuMakanan = new MenuMakanan();  menuMakanan.setVisible(true);  }  });  }  // Variables declaration - do not modify  private javax.swing.JButton addbtn;  private javax.swing.JButton backbtn;  private javax.swing.JButton deletebtn;  private javax.swing.JButton editbtn;  private javax.swing.JLabel jLabel1;  private javax.swing.JPanel jPanel1;  private javax.swing.JPanel jPanel3;  private javax.swing.JScrollPane jScrollPane1;  private javax.swing.JSeparator jSeparator1;  private javax.swing.JSeparator jSeparator2;  private javax.swing.JTable jTable1;  // End of variables declaration  } |

|  |  |
| --- | --- |
| Class TambahMakanan | |
|  | package wanmarized;  import java.sql.Connection;  import java.sql.DriverManager;  import java.sql.PreparedStatement;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.logging.Level;  import java.util.logging.Logger;  import javax.swing.JOptionPane;  import ConfigDatabase.DatabaseConfig;  public class TambahMakanan extends javax.swing.JFrame {  Connection con1;  PreparedStatement insert;  public TambahMakanan() {  initComponents();  setLocationRelativeTo(null);  }  @SuppressWarnings("unchecked")  // Generated Code  private void addbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  if (namaProduk.getText().equals("") || hargaJual.getText().equals("") || hargaModal.getText().equals("")) {  JOptionPane.showMessageDialog(this, "Masih ada kolom yang kosong!");  return;  }  int reply = JOptionPane.showConfirmDialog(null, "Apakah data yang anda masukkan sudah benar?", "Konfirmasi Pendaftaran Akun", JOptionPane.YES\_NO\_OPTION);  if (reply == JOptionPane.NO\_OPTION) {  return;  }  String fieldNamaProduk = namaProduk.getText();  String fieldHargaJual = hargaJual.getText();  String fieldHargaModal = hargaModal.getText();  String query1 = "select \* from produk";  try (Connection connection = new DatabaseConfig().getConnection(); PreparedStatement pst = connection.prepareStatement(query1); ResultSet resultSet = pst.executeQuery()) {  int lastId = 0;  while (resultSet.next()) {  String id = resultSet.getString("id\_produk");  String[] splitId = id.split("D");  lastId = Integer.parseInt(splitId[1]);  }  String query2 = "insert into produk (id\_produk,nama\_produk,harga\_jual, harga\_modal)values(?,?,?,?)";  insert = connection.prepareStatement(query2);  insert.setString(1, "PRD" + (lastId + 1));  insert.setString(2, fieldNamaProduk);  insert.setString(3, fieldHargaJual);  insert.setString(4, fieldHargaModal);  insert.executeUpdate();  JOptionPane.showMessageDialog(this, "Makanan berhasil ditambahkan.");  MenuMakanan menumakanan = new MenuMakanan();  setVisible(false);  dispose();  menumakanan.setVisible(true);  } catch (SQLException ex) {  Logger.getLogger(TambahMakanan.class.getName()).log(Level.SEVERE, null, ex);  }  }  private void backbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  setVisible(false);  dispose();  }  public static void main(String args[]) {  java.awt.EventQueue.invokeLater(new Runnable() {  public void run() {  new TambahMakanan().setVisible(true);  }  });  }  // Variables declaration - do not modify  private javax.swing.JButton addbtn;  private javax.swing.JButton backbtn;  private javax.swing.JTextField hargaJual;  private javax.swing.JTextField hargaModal;  private javax.swing.JLabel jLabel1;  private javax.swing.JLabel jLabel2;  private javax.swing.JLabel jLabel3;  private javax.swing.JLabel jLabel4;  private javax.swing.JPanel jPanel1;  private javax.swing.JPanel jPanel2;  private javax.swing.JTextField namaProduk;  // End of variables declaration  } |

|  |  |
| --- | --- |
| Class EditMakanan | |
|  | package wanmarized;  import java.sql.Connection;  import java.sql.DriverManager;  import java.sql.PreparedStatement;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.logging.Level;  import java.util.logging.Logger;  import javax.swing.JOptionPane;  import javax.swing.table.DefaultTableModel;  import ConfigDatabase.DatabaseConfig;  import wanmarized.MenuMakanan;  public class EditMakanan extends javax.swing.JFrame {  Connection con1;  private String selectedData, selectedProductName, selectedProductSellingPrice, selectedProductCapitalPrice;  public EditMakanan(String selectedData, String selectedProductName, String selectedProductSellingPrice, String selectedProductCapitalPrice) {  initComponents();  setLocationRelativeTo(null);  this.selectedData = selectedData;  this.selectedProductName = selectedProductName;  this.selectedProductSellingPrice = selectedProductSellingPrice;  this.selectedProductCapitalPrice = selectedProductCapitalPrice;  this.deriveSelectedProductData();  }  @SuppressWarnings("unchecked")  // Generated Code  private void deriveSelectedProductData() {  namaProduk.setText(selectedProductName);  hargaJual.setText(selectedProductSellingPrice);  hargaModal.setText(selectedProductCapitalPrice);  }  private void editbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  if (namaProduk.getText().equals(this.selectedProductName) && hargaJual.getText().equals(this.selectedProductSellingPrice) && hargaModal.getText().equals(this.selectedProductCapitalPrice)) {  JOptionPane.showMessageDialog(this, "Anda belum melakukan perubahan data.");  return;  }  String query = "update produk set nama\_produk= ?,harga\_jual= ?,harga\_modal= ? where id\_produk= ?";  try (Connection connection = new DatabaseConfig().getConnection(); PreparedStatement pst = connection.prepareStatement(query)) {  String query2 = "select \* from produk where id\_produk= ?";  try (Connection connection2 = new DatabaseConfig().getConnection(); PreparedStatement pst2 = connection2.prepareStatement(query2)) {  pst2.setString(1, this.selectedData);  try (ResultSet resultSet = pst2.executeQuery()) {  pst.setString(1, (namaProduk.getText().equals("")) ? resultSet.getString("nama\_produk") : namaProduk.getText());  pst.setString(2, (hargaJual.getText().equals("")) ? resultSet.getString("harga\_jual") : hargaJual.getText());  pst.setString(3, (hargaModal.getText().equals("")) ? resultSet.getString("harga\_modal") : hargaModal.getText());  }  }  pst.setString(4, this.selectedData);  pst.executeUpdate();  JOptionPane.showMessageDialog(this, "Perubahan yang anda lakukan berhasil disimpan.");  MenuMakanan menumakanan = new MenuMakanan();  setVisible(false);  dispose();  menumakanan.setVisible(true);  } catch (SQLException ex) {  Logger.getLogger(EditMakanan.class.getName()).log(Level.SEVERE, null, ex);  }  }  private void backbtnActionPerformed(java.awt.event.ActionEvent evt) {  // TODO add your handling code here:  setVisible(false);  dispose();  }  // Variables declaration - do not modify  private javax.swing.JButton backbtn;  private javax.swing.JButton editbtn;  private javax.swing.JTextField hargaJual;  private javax.swing.JTextField hargaModal;  private javax.swing.JLabel jLabel1;  private javax.swing.JLabel jLabel2;  private javax.swing.JLabel jLabel3;  private javax.swing.JLabel jLabel4;  private javax.swing.JPanel jPanel1;  private javax.swing.JPanel jPanel2;  private javax.swing.JTextField namaProduk;  // End of variables declaration  } |

2.7 Screenshot Program yang Dijalankan