

**LAPORAN TUGAS MATA KULIAH
PEMROGRAMAN BERORIENTASI OBYEK**

Dosen Pengampu:

Bayu Adhi Nugroho, Ph.D.



Disusun Oleh:

Muhammad Yusuf Multazam

09020624053

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUNAN AMPEL SURABAYA
TAHUN 2025**

A. Latar Belakang

Dalam proses pengembangan perangkat lunak, sistem login merupakan komponen dasar yang berfungsi untuk mengontrol akses pengguna terhadap suatu aplikasi. Dengan adanya sistem login, hanya pengguna yang terdaftar dan memiliki kredensial yang valid yang dapat mengakses fitur-fitur tertentu.

Pada praktikum ini dibuat sebuah aplikasi **Login Form** menggunakan **Java Swing** di **NetBeans IDE** dengan koneksi ke **database MySQL** melalui **Java Persistence API (JPA)**. JPA memudahkan pengembang dalam melakukan operasi **CRUD (Create, Read, Update, Delete)** terhadap database tanpa perlu menulis banyak perintah SQL secara manual, karena seluruh proses interaksi data dikelola melalui **entity class** dan **EntityManager**.

Aplikasi ini tidak hanya menyediakan fitur **login dan registrasi pengguna**, tetapi juga **lupa password (forgot password)** dan **reset password** untuk meningkatkan keamanan dan kemudahan pengguna.

B. Tujuan

1. Mempelajari cara membuat **form login** berbasis Java Swing.
2. Menerapkan **Java Persistence API (JPA)** untuk menghubungkan aplikasi dengan database MySQL.
3. Membuat entity class untuk merepresentasikan tabel database dalam bentuk objek Java.
4. Membuat fitur **login, register, forgot password, dan reset password** dengan transaksi JPA.
5. Melatih kemampuan memahami konsep **EntityManager, Persistence Unit, dan ORM (Object Relational Mapping)**.
6. Membuat button download yang digunakan untuk mendownload data pada tabel

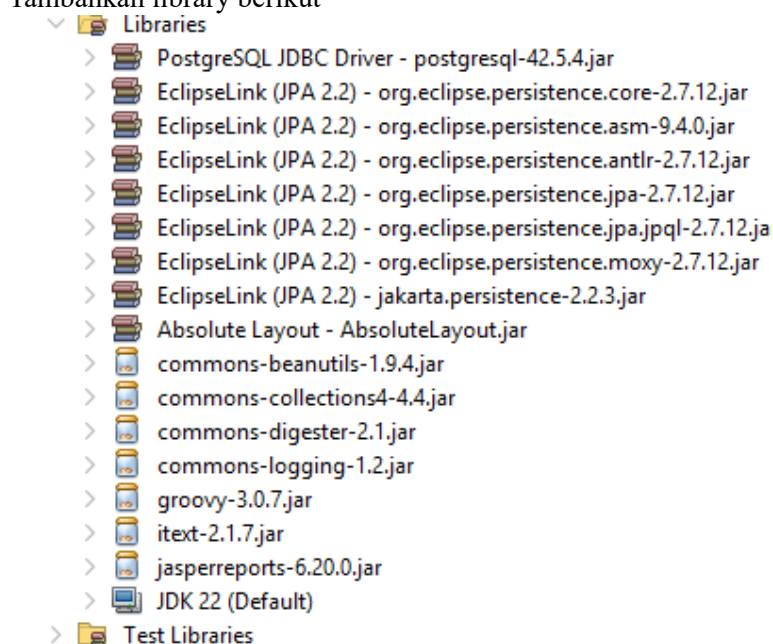
C. Implementasi

Langkah – Langkah :

1. Buat database baru terlebih dahulu
2. Kemudian buat table berikut pada database

```
1 CREATE TABLE login (  
2     username VARCHAR(50) PRIMARY KEY,  
3     password VARCHAR(255) NOT NULL,  
4     security_question VARCHAR(255) NOT NULL,  
5     security_answer VARCHAR(255) NOT NULL  
6 );  
7 CREATE TABLE menu (  
8     id_menu INT PRIMARY KEY,  
9     nama_menu VARCHAR(100) NOT NULL,  
10    kategori VARCHAR(50) NOT NULL,  
11    harga INT NOT NULL  
12 );  
13  
14 CREATE TABLE pesanan (  
15     id_pesanan INT PRIMARY KEY,  
16     nama_pelanggan VARCHAR(100) NOT NULL,  
17     tanggal TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
18 );  
19  
20 CREATE TABLE detail_pesanan (  
21     id_detail INT PRIMARY KEY,  
22     id_pesanan INT REFERENCES pesanan(id_pesanan) ON DELETE CASCADE,  
23     id_menu INT REFERENCES menu(id_menu) ON DELETE CASCADE,  
24     jumlah INT NOT NULL,  
25     subtotal INT NOT NULL  
26 );
```

3. Buka neatbeans dan buat project baru
4. Kemudian buat koneksi persistence untuk menghubungkan ke postgre
5. Tambahkan library berikut



6. Buat 4 jframe form, untuk login, register, forgot password, dan reset password
7. Kemudian design ke empat jframe form nya

Login :



A login form with a blue background and white text. The title "LOGIN" is centered at the top in a large, bold, serif font. Below the title, there are two input fields: "Username" and "Password", both with white text and white input boxes. Below the password field, there is a link "Forgot Password?" in white text. At the bottom, there is a "Login" button with white text on a blue background, and a link "Belum Punya Akun?" in white text below it.

Register :



A register form with a blue background and white text. The title "REGISTER" is centered at the top in a large, bold, serif font. Below the title, there are four input fields: "Username", "Password", "Pertanyaan Keamanan", and "Jawaban", all with white text and white input boxes. The "Pertanyaan Keamanan" field has a dropdown menu with the text "Tempat liburan pertama keluarga?" and a downward arrow. At the bottom right, there is a "Submit" button with white text on a blue background.

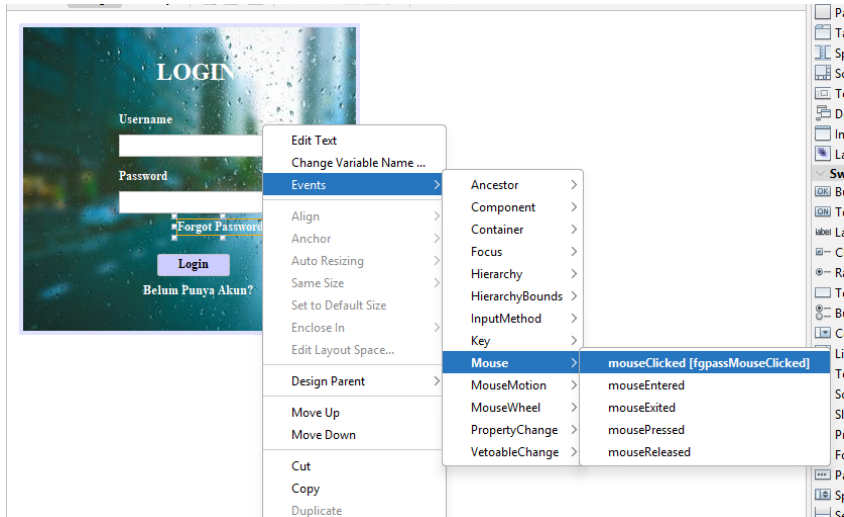
Forgor Password :



A forgot password form with a blue background and white text. The title "FORGOT PASSWORD" is centered at the top in a large, bold, serif font. Below the title, there are three input fields: "Username", "Pertanyaan Keamanan", and "Jawaban", all with white text and white input boxes. The "Username" field has a "Cari Username" button with white text on a blue background. The "Pertanyaan Keamanan" field has a dropdown menu with the text "Tempat liburan pertama keluarga?" and a downward arrow. At the bottom right, there is a "Verifikasi" button with white text on a blue background.

Reset Password :

8. Pada bagian Login agar tulissan “Forgot password” dan “Belum Punya Akun” bisa di klik lakukan cara ini



Setelah itu tambahkan source code berikut pada masing masing tulisan

```
private void fgpassMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    Pass lupa = new Pass();
    lupa.setVisible(true);
    this.dispose();
    setLocationRelativeTo(null);
}

private void regisMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    Register regis = new Register();
    regis.setVisible(true);
    this.dispose();
}
```

9. Kemudian ketik source code berikut untuk setiap JFrame Login :

```

private void bloginActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    EntityManagerFactory emf = Persistence.createEntityManagerFactory("Pertemuan14PU");
    EntityManager em = emf.createEntityManager();

    try {
        String username = tfuser.getText();
        String password = new String(tfpass.getPassword());

        Pertemuan14.Login user = em.find(Pertemuan14.Login.class, username);

        if (user != null && user.getPassword().equals(password)) {
            JOptionPane.showMessageDialog(this, "Login berhasil!");
            // TODO: pindah ke halaman berikutnya
            Cafeku c = new Cafeku();
            c.setVisible(true);
            this.dispose();
        } else {
            JOptionPane.showMessageDialog(this, "Username atau Password salah!");
        }
    } finally {
        em.close();
        emf.close();
    }
}

```

Register :

```

private void bsubmitActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    EntityManagerFactory emf = Persistence.createEntityManagerFactory("Pertemuan14PU");
    EntityManager em = emf.createEntityManager();

    try {
        String username = tfuser.getText();
        String password = new String(tfpass.getPassword());
        String pertanyaan = cbpertanyaan.getSelectedItemAt().toString(); // ComboBox pertanyaan
        String jawaban = tfjawaban.getText(); // TextField jawaban

        // Cek apakah username sudah ada
        Pertemuan14.Login existingUser = em.find(Pertemuan14.Login.class, username);
        if (existingUser != null) {
            JOptionPane.showMessageDialog(this, "Username sudah digunakan!");
            return;
        }

        // Buat object baru untuk register
        Pertemuan14.Login newUser = new Pertemuan14.Login();
        newUser.setUsername(username);
        newUser.setPassword(password);
        newUser.setSecurityQuestion(pertanyaan);
        newUser.setSecurityAnswer(jawaban);
    }
}

```

```

        // Simpan ke database
        em.getTransaction().begin();
        em.persist(newUser);
        em.getTransaction().commit();

        JOptionPane.showMessageDialog(this, "Pendaftaran berhasil!");

        // Pindah ke frame login
        Loginn l = new Loginn();
        l.setVisible(true);
        this.dispose();

    } finally {
        em.close();
        emf.close();
    }

}

```

Forgot Password :

```

private void bsubmitActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    EntityManagerFactory emf = Persistence.createEntityManagerFactory("Pertemuan14PU");
    EntityManager em = emf.createEntityManager();

    try {
        String username = tfuser.getText();
        String password = new String(tfpass.getPassword());
        String pertanyaan = cbpertanyaan.getSelectedItem().toString(); // ComboBox pertanyaan
        String jawaban = tfjawaban.getText(); // TextField jawaban

        // Cek apakah username sudah ada
        Pertemuan14.Login existingUser = em.find(Pertemuan14.Login.class, username);
        if (existingUser != null) {
            JOptionPane.showMessageDialog(this, "Username sudah digunakan!");
            return;
        }

        // Buat object baru untuk register
        Pertemuan14.Login newUser = new Pertemuan14.Login();
        newUser.setUsername(username);
        newUser.setPassword(password);
        newUser.setSecurityQuestion(pertanyaan);
        newUser.setSecurityAnswer(jawaban);
    }
}

```

```

private void bverifActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String answer = tfjawaban.getText().trim();
    if (answer.isEmpty()) {
        JOptionPane.showMessageDialog(this, "Masukkan jawaban!");
        return;
    }

    if (answer.equals(currentUser.getSecurityAnswer())) {
        JOptionPane.showMessageDialog(this, "Jawaban benar. Silakan ubah password.");
        Resetpass resetFrame = new Resetpass(currentUser.getUsername());
        resetFrame.setVisible(true);
        this.dispose();
    } else {
        JOptionPane.showMessageDialog(this, "Jawaban salah!");
    }
}

```

Reset Password :

```

private void bresetActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String newPass1 = new String(tfpass1.getPassword().trim());
    String newPass2 = new String(tfpass2.getPassword().trim());

    if (newPass1.isEmpty() || newPass2.isEmpty()) {
        JOptionPane.showMessageDialog(this, "Password tidak boleh kosong!");
        return;
    }

    if (!newPass1.equals(newPass2)) {
        JOptionPane.showMessageDialog(this, "Password tidak cocok!");
        return;
    }

    EntityManager em = emf.createEntityManager();
    try {
        Login user = em.find(Login.class, username);

        if (user != null) {
            em.getTransaction().begin();
            user.setPassword(newPass1);
            em.getTransaction().commit();

            JOptionPane.showMessageDialog(this, "Password berhasil diubah!");
            this.dispose();
            new Loginn().setVisible(true); // ganti dengan JFrame login kamu
        } else {
            JOptionPane.showMessageDialog(this, "Password gagal diubah!");
        }
    } finally {
        em.close();
    }
}

```

Tambah variable dan parameter berikut

```

private String username;

```



```

public Resetpass(String username) {
    initComponents();
    this.username = username;
    setLocationRelativeTo(null);
}

```

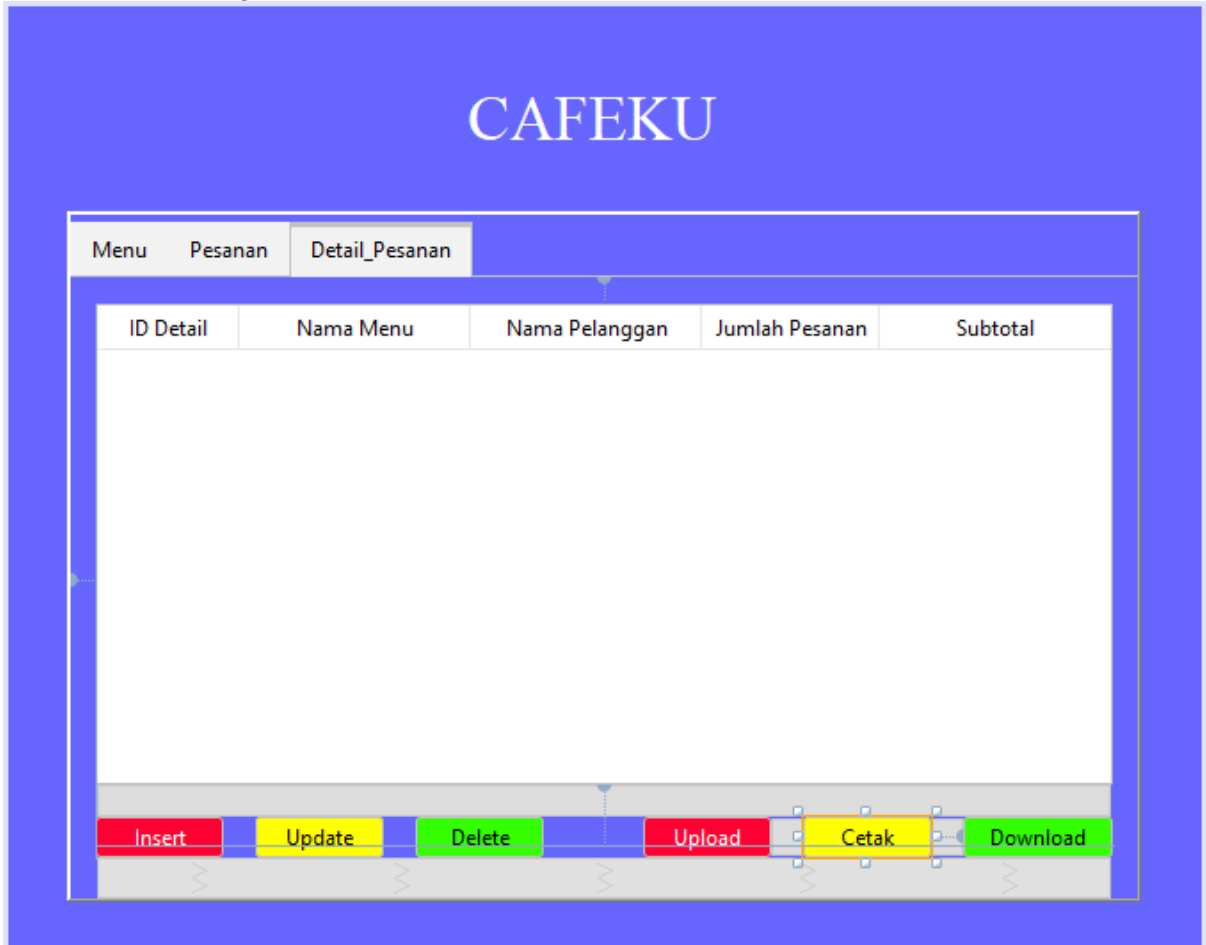
10. Import berikut pada setiap class JFrame form

```

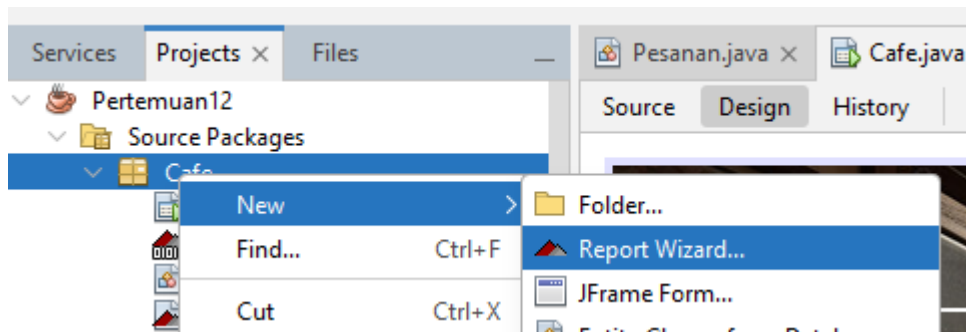
import javax.persistence.*;
import javax.swing.JOptionPane;

```

11. Kemudian buat design untuk 3 tabel relasi



12. Buat 9 dialog, 3 untuk masing masing table yaitu insert, update, dan delete kemudian masukkan source code nya untuk crud
13. Buat juga button upload untuk setiap table, setelah membuat masukkan source code nya untuk upload
14. Buat button cetak untuk setiap table
15. Tambahkan juga class untuk jasper yang digunakan untuk mencetak laporan di bagian berikut. buat 3 yang digunakan untuk table menu, pesanan, dan detailpesanan



16. Kemudian design bentuk laporan jasper untuk masing masing tabel
17. Masukkan code berikut untuk cetak di masing masing table sesuaikan nama file jrxml nya

```
private void jButton10ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try {
        // Lokasi file laporan .jasper (hasil compile dari .jrxml)
        String reportPath = "src/Pertemuan14/Menu.jasper";
        Connection conn = getConnection();

        net.sf.jasperreports.engine.JasperPrint jp
            = net.sf.jasperreports.engine.JasperFillManager.fillReport(reportPath, null, conn);

        net.sf.jasperreports.view.JasperViewer.viewReport(jp, false);
    } catch (JRException e) {
        javax.swing.JOptionPane.showMessageDialog(null, "Gagal mencetak laporan: " + e.getMessage());
    }
}
```

18. Buat juga untuk button download untuk setiap table

Contoh source code untuk table menu

```
private void jButton9ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    JFileChooser chooser = new JFileChooser();
    chooser.setDialogTitle("Simpan Data Menu sebagai CSV");

    int result = chooser.showSaveDialog(this);

    if (result == JFileChooser.APPROVE_OPTION) {
        File file = chooser.getSelectedFile();
        if (!file.getName().toLowerCase().endsWith(".csv")) {
            file = new File(file.getAbsolutePath() + ".csv");
        }

        if (file.exists()) {
            int confirm = JOptionPane.showConfirmDialog(
                this,
                "File sudah ada, apakah ingin menimpa?",
                "Konfirmasi",
                JOptionPane.YES_NO_OPTION
            );

            if (confirm != JOptionPane.YES_OPTION) {
                return;
            }
        }

        EntityManagerFactory emf = null;
        EntityManager em = null;
        FileWriter fw = null;
    }
}
```

```

try {
    emf = Persistence.createEntityManagerFactory("Pertemuan14PU");
    em = emf.createEntityManager();
    List<Menu> dataMenu = em.createQuery("SELECT m FROM Menu m", Menu.class).getResultList();

    // ----- TULIS FILE CSV -----
    fw = new FileWriter(file);

    for (Menu m : dataMenu) {
        fw.append(m.getIdMenu() + ";");
        fw.append(m.getNamaMenu() + ";");
        fw.append(m.getKategori() + ";");
        fw.append(m.getHarga() + "\n");
    }

    fw.flush();

    JOptionPane.showMessageDialog(this, "Download CSV Menu Berhasil!");
} catch (HeadlessException | IOException e) {
    JOptionPane.showMessageDialog(this, "Gagal menyimpan file: " + e.getMessage());
} finally {
    try {
        if (fw != null) {
            fw.close();
        }
    } catch (IOException ex) {
    }

    if (em != null) {
        em.close();
    }

    if (emf != null) {
        emf.close();
    }
}
}

```

19. Masukkan juga source code nya untuk table pesanan dan detail pesanan sesuaikan programnya dengan masing masing table
20. Setelah selesai program bisa dijalankan dari login dan akan terhubung ke cafe