**PENJELASAN KODE BACKEND**

Nama: Richard Accita Sistwanto

NRP: C14220059

1. ***Login.php***
2. Mendapatkan data dari request

- Mengambil data username dan password dari request PHP (POST) yang dikirim oleh user

B. Proses verifikasi login

- Mengatur koneksi ke local database (localhost) yang dimana dikoneksikan melalui function/class “mysqli” dan diberikan pesan jika koneksi gagal

C. Menjalankan query untuk mendapatkan daftar username dan password

- Disini akan dijalankan query dengan Bahasa MySQL untuk mendapatkan list data yang diinginkan dan menjalankan program itu lewat koneksi yang tadi sudah dibuat dan menampung hasilnya di variabel “$result”

D. Memeriksa hasil query

- Pada bagian ini akan diperiksa apakah hasil dari query tersebut terdapat baris atau tidak. Jika ya, maka akan membaca isi data dari setiap row menggunakan fungsi “fetch\_assoc” dan jika username dan password hasil request POST sama dengan data yang ada di dalam hasil query yang ada maka variabel “$loginSuccess” diubah valuenya menjadi true. Jika tidak, maka akan di print di console “Table tidak memiliki isi”.

E. Mengirimkan respon

- Pada bagian ini, variabel “$loginSuccess” apakah true or false. Jika true, maka pesan respon akan diteruskan ke user dengan format JSON dengan tulisan “Login successful”. Jika false, maka pesan respon akan diteruskan ke user adalah “Login failed”

F. Menutup koneksi

- Seperti namanya, pada bagian ini koneksi kita ke database akan ditutup

1. ***Signup.php***
2. Mendapatkan data dari request:

- Mengambil nilai username dan password dari request POST yang dikirim oleh user.

B. Proses verifikasi login:

- Bagian ini mengatur koneksi ke database MySQL menggunakan objek mysqli. Anda perlu mengganti nilai $user, $pass, dan $dbname dengan informasi login database Anda. Jika koneksi gagal, pesan "Koneksi gagal" akan ditampilkan dan skrip akan berhenti.

C. Melakukan pendaftaran/memasukkan akun ke dalam database:

- Di bagian ini, program akan menjalankan query sql “INSERT” digunakan untuk memasukkan data username dan password yang diperoleh dari request POST ke dalam tabel anggota\_table. Hasil query tidak disimpan dalam variabel, karena tujuan utama adalah memasukkan data baru ke dalam tabel.

D. Memeriksa hasil query:

- Setelah query dijalankan, kode memeriksa apakah query berhasil dijalankan atau tidak dengan menggunakan “$conn->query($sql)”. Jika query berhasil dijalankan, variabel “$signupSuccess” diubah menjadi true. Jika gagal, “$signupSuccess” diubah menjadi false.

E. Mengirimkan respon:

- Berdasarkan nilai “$signupSuccess”, pesan respon yang sesuai dibentuk dalam bentuk array asosiatif. Pesan respon tersebut dikodekan dalam format JSON menggunakan fungsi “json\_encode()”. Pesan respon kemudian dicetak ke output.

F. Menutup koneksi:

- Seperti namanya, pada bagian ini koneksi kita ke database akan ditutup

1. ***getDataByNRP.php***

A. Mendapatkan data dari request:

- Dalam bagian ini, kita mengambil data NRP dari permintaan (request) yang dikirim oleh pengguna menggunakan metode GET dalam PHP. Nilai NRP ini akan digunakan untuk mencari data mahasiswa yang terkait.

B. Proses verifikasi login

- Pada bagian ini, kita mengatur koneksi ke local database (localhost) menggunakan objek mysqli dalam PHP. Jika koneksi gagal, maka akan ditampilkan pesan “Koneksi gagal" dengan informasi kesalahan yang diperoleh dari “connect\_error”.

C. Menjalankan query untuk mendapatkan data-data anggota:

- Bagian ini menjalankan query SQL untuk mengambil semua data dari tabel "anggota\_table" dalam database. Hasil query disimpan dalam variabel “$result” yang akan digunakan untuk memeriksa apakah terdapat baris data yang ditemukan.

D. Memeriksa hasil query:

- Pada bagian ini, kita memeriksa apakah hasil query menghasilkan baris data atau tidak. Jika jumlah baris (rows) yang ditemukan lebih dari 0, maka kita melakukan loop melalui setiap baris data yang ditemukan menggunakan “fetch\_assoc()”. Data dari setiap baris tersebut disimpan dalam array “$mahasiswa”. Jika hasil query tidak menghasilkan baris data, maka akan dicetak pesan "Table tidak memiliki isi".

E. Mencari data mahasiswa berdasarkan NRP dan mengirimkan respon:

- Setelah mendapatkan data mahasiswa dan melakukan pengecekan, kita mencari data mahasiswa yang sesuai dengan NRP yang diberikan. Jika ditemukan, data mahasiswa tersebut akan dikirim sebagai respons dalam format JSON menggunakan fungsi “json\_encode()”. Jika tidak ditemukan, maka respons akan berupa JSON kosong.

F. Menutup koneksi:

- Seperti namanya, pada bagian ini koneksi kita ke database akan ditutup

1. ***putPatch.php***

A. Mendapatkan data dari request:

- Kode ini mengambil data nrp dan ipk dari request PHP (POST) yang dikirim oleh pengguna.

B. Proses verifikasi login:

- Kode ini mengatur koneksi ke database lokal (localhost) menggunakan objek mysqli. Jika koneksi gagal, program akan berhenti dan menampilkan pesan kesalahan.

C. Menjalankan query untuk mengganti anggota yang memiliki nrp yang diberi user dengan ipk yang baru:

- Pada bagian ini, query SQL dijalankan untuk memperbarui nilai IPK mahasiswa dalam database berdasarkan NRP. Query tersebut menggunakan tabel “anggota\_table” dalam database.

D. Memeriksa hasil query:

- Setelah menjalankan query, hasilnya disimpan dalam variabel “$result”. Selanjutnya, kode ini memeriksa apakah query berhasil dijalankan atau tidak dengan membandingkan nilai dari “$conn->query($sql)”. Jika nilainya TRUE, value variabel “$ipkUpdated” diubah menjadi true.

E. Mengirimkan respon:

- Berdasarkan nilai variabel “$ipkUpdated”, kode ini membuat pesan respon yang akan dikirimkan ke pengguna. Pesan tersebut dikodekan dalam format JSON menggunakan fungsi “json\_encode()” dan di-print ke user.

F. Menutup koneksi:

- Seperti namanya, pada bagian ini koneksi kita ke database akan ditutup