

1. User mengakses halaman utama/login SIAM User memulai dengan membuka browser dan memasukkan URL halaman utama Sistem Informasi Akademik Mahasiswa (SIAM) di address bar. Ini akan memuat halaman login SIAM.
2. Login menggunakan HTML dan CSS: Pada halaman login, HTML digunakan untuk membuat elemen-elemen seperti formulir login, tombol, dan teks input untuk username dan password. CSS digunakan untuk mengatur tata letak dan tampilan visual halaman agar terlihat menarik dan mudah digunakan.
3. Validasi login menggunakan PHP: Ketika user memasukkan username dan password dan menekan tombol login, informasi tersebut dikirim ke server menggunakan bahasa pemrograman PHP. PHP kemudian memproses data tersebut, memeriksa kebenaran kredensial di database. Jika username dan password cocok, PHP membuat sesi baru untuk user tersebut dan mengirimkan balikan ke browser, memberitahu bahwa user telah berhasil login.
4. Pengaturan tampilan menggunakan CSS: Setelah login berhasil, user diarahkan ke halaman KRS. HTML digunakan untuk membangun struktur halaman KRS, sementara CSS mengatur tampilan dan tata letaknya. Ini termasuk hal-hal seperti ukuran teks, warna latar belakang, dan posisi elemen-elemen halaman.
5. **\*\*Interaksi dinamis dengan JavaScript (opsional)\*\***: JavaScript dapat digunakan untuk menambahkan interaktivitas ke halaman KRS. Misalnya, validasi formulir, efek animasi, atau pembaruan dinamis halaman tanpa harus me-refresh seluruh halaman.
6. **\*\*Pengambilan data dari database menggunakan PHP**: PHP akan mengambil data KRS mahasiswa dari database. Ini melibatkan penggunaan bahasa SQL untuk mengambil data yang sesuai dari tabel database.
7. Pengiriman data ke halaman HTML: Setelah PHP mendapatkan data KRS dari database, itu akan menyusunnya ke dalam format HTML. Data ini kemudian disematkan ke dalam halaman HTML yang sudah ada.
8. Penggunaan session/cookie: PHP menggunakan sesi untuk mempertahankan informasi login pengguna selama mereka menjelajahi halaman. Cookie digunakan untuk menyimpan ID sesi pada browser pengguna sehingga server dapat mengenali pengguna tersebut dari permintaan ke permintaan.
9. Pengiriman halaman KRS ke browser: Setelah proses di atas selesai, halaman KRS yang lengkap telah disiapkan dalam bentuk HTML. Server kemudian mengirimkan halaman HTML ini kembali ke browser pengguna melalui koneksi HTTP.
10. Tampilan halaman KRS di jendela browser: Akhirnya, browser menerima halaman HTML dari server dan menampilkan konten tersebut di jendela browser pengguna. Pengguna sekarang dapat melihat dan berinteraksi dengan halaman KRS mereka, dengan data yang ditampilkan sesuai dengan informasi yang diambil dari database.

