

Nama : Marvin Gilbrand Adu

Nim : 2201010598

Proyek: Simulator Penjadwalan Proses

Gambaran Proyek

Simulator Penjadwalan Proses akan menjadi aplikasi yang mensimulasikan berbagai algoritma penjadwalan CPU seperti First-Come-First-Serve (FCFS), Shortest Job First (SJF), Round Robin (RR), dan Priority Scheduling. Aplikasi ini akan memungkinkan pengguna untuk memasukkan proses dengan berbagai atribut (nama, waktu kedatangan, waktu burst, dll.) dan kemudian mensimulasikan proses penjadwalan, menunjukkan urutan eksekusi dan menghitung metrik kunci seperti waktu tunggu, waktu turnaround, dan waktu penyelesaian.

Fitur

1. Input Proses:

- Memungkinkan pengguna untuk memasukkan detail proses seperti nama proses, waktu kedatangan, waktu burst, dan prioritas (jika ada).
- Menyediakan opsi untuk menambah, mengedit, dan menghapus proses.

2. Pemilihan Algoritma:

- Menyediakan dropdown atau menu pilihan untuk memilih algoritma penjadwalan (FCFS, SJF, RR, Prioritas).
- Memungkinkan pengguna untuk mengatur parameter untuk algoritma, seperti quantum waktu untuk Round Robin.

3. Simulasi:

- Mensimulasikan proses penjadwalan berdasarkan algoritma yang dipilih.
- Menampilkan diagram Gantt atau timeline yang menunjukkan urutan eksekusi proses.
- Menghitung dan menampilkan metrik kunci seperti waktu tunggu, waktu turnaround, dan waktu penyelesaian untuk setiap proses.

4. Visualisasi:

- Menggunakan elemen grafis untuk memvisualisasikan penjadwalan dan eksekusi proses.
- Menyediakan antarmuka yang jelas dan ramah pengguna untuk melihat dan berinteraksi dengan simulasi.

5. Laporan:

- Menghasilkan laporan yang merangkum hasil simulasi, termasuk rata-rata waktu tunggu, rata-rata waktu turnaround, dan statistik relevan lainnya.
- Memungkinkan pengguna untuk menyimpan atau mengekspor laporan.

Langkah Implementasi

1. Definisikan Objek Proses:

- Gunakan kelas ProcessObj yang Anda berikan sebagai dasar untuk menyimpan informasi proses.

2. Antarmuka Pengguna:

- Kembangkan antarmuka yang ramah pengguna untuk memasukkan detail proses dan memilih algoritma.
- Gunakan Java Swing atau JavaFX untuk aplikasi desktop, atau Android Studio untuk aplikasi seluler.

3. Algoritma Penjadwalan:

- Implementasikan algoritma penjadwalan yang berbeda sebagai metode atau kelas terpisah.
- Pastikan setiap algoritma menghitung metrik yang diperlukan dengan benar dan mensimulasikan proses penjadwalan.

4. Visualisasi:

- Gunakan komponen grafis untuk memvisualisasikan proses penjadwalan (misalnya, diagram Gantt).
- Perbarui UI secara real-time untuk mencerminkan simulasi.

5. Pengujian dan Debugging:

- Uji aplikasi dengan berbagai input dan kasus edge untuk memastikan akurasi dan keandalan.
- Debug masalah yang muncul selama simulasi.