

SIMULADOR WEB DE UM SISTEMA OPERACIONAL PARA FINS EDUCACIONAIS

TIAGO DE F. FARIA¹, MURILO DA S. DANTAS²

¹ Graduando em Tecnologia de Análise e Desenvolvimento de Sistemas, IFSP, Campus Jacareí, tiago.faria@aluno.ifsp.edu.br.

² Docente do curso Análise e Desenvolvimento de Sistemas, IFSP, Campus Jacareí, murilo.dantas@ifsp.edu.br.

Área de conhecimento (Tabela CNPq): 1.03.03.04-9 Sistemas de Informação

RESUMO: O ensino-aprendizagem da estrutura e do funcionamento de um sistema operacional envolve conceitos e algoritmos complexos e de difícil entendimento, o que acaba exigindo muito esforço de abstração do aluno. O sistema operacional, por ser o principal software de um computador, é responsável por muitas tarefas, e pode ter seu aprendizado simplificado através de simulações que demonstrem seu funcionamento de forma visual. Baseado nisso, foi proposto o desenvolvimento de um ambiente na web capaz de simular e demonstrar algumas das principais funcionalidades de um sistema operacional. Esta ferramenta poderá auxiliar no processo de aprendizado do aluno e facilitar a compreensão de diversos conceitos.

PALAVRAS-CHAVE: Escalonamento de CPU; Sincronismo; Deadlocks; Memória; Aprendizado.

WEB SIMULATOR OF AN OPERATING SYSTEM FOR EDUCATIONAL PURPOSES

ABSTRACT: The teaching/learning of the structure and functioning of an operating system involves complex concepts and algorithms that are difficult to understand, which ends up demanding a lot of abstraction effort from the student. The operating system, being the main software of a computer, is responsible for many tasks, and may have its learning simplified through simulations that demonstrate its operation visually. Based on that, it was proposed to develop a web environment capable of simulating and demonstrating some of the main functionalities of an operating system. This tool will assist in the student's learning process and facilitate the understanding of several concepts.

KEYWORDS: CPU Scheduling; Synchronism; Deadlocks; Memory; Learning.

INTRODUÇÃO

Um sistema operacional pode ser definido como um programa, ou um conjunto deles, responsável por exercer diversas funções, como o gerenciamento dos recursos físicos e pela realização da interface entre usuário e máquina (BOS e TANENBAUM, 2016). O funcionamento das diversas tarefas envolvidas na gerência dos recursos utiliza conceitos abstratos, o que dificulta sua compreensão. Diversas tarefas, tais como o escalonamento de CPU, sincronismo entre processos, tratamento e recuperação de deadlocks, e gerência da memória principal, são executadas pelo sistema operacional, e envolvem algoritmos complexos, que podem não ser completamente compreendidos pelos estudantes, se não demonstradas de forma prática.

Tornar o processo de ensino-aprendizagem mais construtivo, facilita a absorção do conhecimento, pois o estudante terá acesso a um ensino mais interativo e mais atrativo (LOIOLA, 2009). Neste processo, há muitas iniciativas, sendo que uma se destaca: os ambientes simulados (Moraes et al, 1111). Em ambientes simulados, o principal objetivo é simular os principais algoritmos de sistemas operacionais, demonstrando seu funcionamento, gerando análises gráficas e animações interativas (MAZIERO, 2002). Tendo isso em mente, foi proposto o desenvolvimento de um sistema web, cujo

objetivo principal é simular a execução de algoritmos utilizados nas principais funcionalidades do sistema operacional, e que auxilie no ensino-aprendizagem destas.

MATERIAL E MÉTODOS

Equipamento

O dispositivo utilizado para desenvolvimento deste projeto foi um notebook da marca Lenovo. O dispositivo possui o processador Intel Core i5 da 8ª geração, 8GB de memória RAM, HD de 1TB e a placa de vídeo NVIDIA GeForce MX110 2GB.

Scrum

O projeto está sendo desenvolvido utilizando o Scrum, um framework para desenvolver, entregar e manter produtos complexos (SCHWABER e SUTHERLAND, 2017). Dentro do ambiente Scrum existem diversos papéis, eventos e regras. O Scrum facilita o tratamento e a resolução de problemas complexos e aumenta a produtividade e criatividade, auxiliando na criação de um produto de alto valor. Sua abordagem é iterativa e incremental, para tornar o processo mais previsível e controlar riscos.

No Scrum existem três pilares que auxiliam no controle de processos, sendo: transparência entre todos os envolvidos no desenvolvimento do produto; inspeção de artefatos e do progresso rumo ao objetivo; e adaptação a possíveis desvios durante o caminho.

Desenvolvimento Web

Um website é formado por documentos de texto, cores, links, imagens e formatação que permite a navegação dos usuários através do seu acesso pela sua URL (Uniform Resource Locator) em um navegador (MADUREIRA, 2017). O desenvolvimento de um site pode passar por diversas etapas até estar completo e pronto para ser utilizado. Entre essas possíveis etapas pelas quais um site passa estão o planejamento, a prototipação, a codificação e a realização de testes.

Durante o planejamento, foram criados wireframes (esboços) de cada página do site, e os storyboards, que definem o fluxo de informações entre as páginas. A prototipação parte de tudo que foi definido no planejamento, para a construção da arquitetura do site. A codificação teve início quando todo o protótipo e o layout do site estavam prontos. Toda a estrutura do site foi feita em HTML (Hypertext Markup Language) e os estilos utilizados foram criados em arquivos CSS (Cascading Style Sheets). Outras funcionalidades do site foram escritas em JavaScript ou Python.

Framework Django

A linguagem de programação Python foi criada em 1991, por Guido van Rossum, com o intuito de ser fácil de aprender, e voltado para pessoas com pouca experiência em programação. A linguagem possui suporte a módulos e frameworks desenvolvidos pela comunidade, que são bibliotecas com funções especializadas. Entre esses módulos e frameworks existem alguns voltados ao desenvolvimento web, como é o caso do Django.

O Django é um framework de alto nível que trabalha com o tratamento de requisições, mapeamento objeto-relacional e preparação de respostas HTTP (RAMOS, 2018), por exemplo. O framework Django foi utilizado para funções específicas neste projeto, relacionadas à parte de back-end do site, para fazer o relacionamento com o banco de dados.

RESULTADOS E DISCUSSÃO

O desenvolvimento do sistema se iniciou com as definições básicas do sistema, incluindo a escolha de um nome e das linguagens e ferramentas utilizadas. Como nome do sistema, foi escolhido “SONaWeb” (Sistema Operacional na Web). Como mencionado anteriormente, para o desenvolvimento do sistema está sendo utilizado o framework Django, que segue um padrão de projetos, em camadas, denominado MTV (Model-Template-View). A arquitetura do sistema SONaWeb, que pode ser vista na Figura 1, por ser baseado nesse framework, deve obedecer a certas regras para que as camadas funcionem bem, em conjunto.

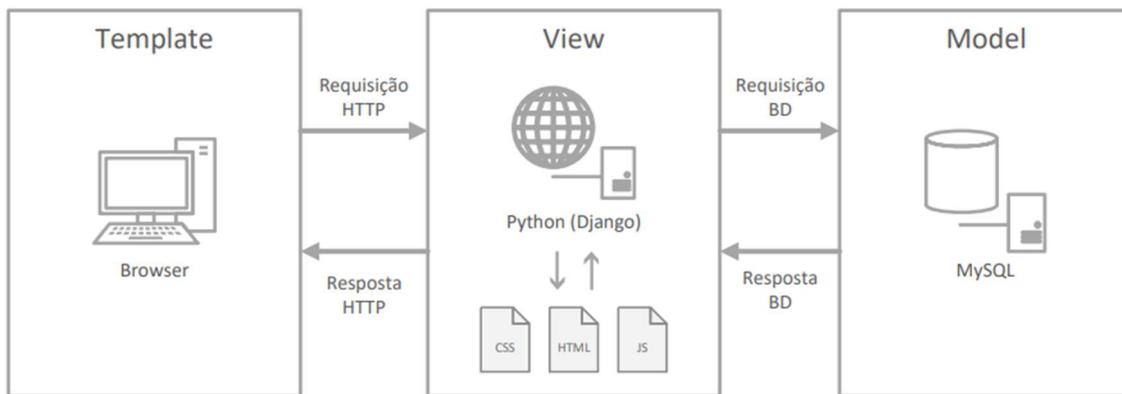


FIGURA 1. Arquitetura do sistema SOnaWeb.

Para a simulação do Escalonamento de CPU, o usuário deve escolher um algoritmo, informar a quantidade de processos na simulação e seus respectivos tempos de burst e de chegada (veja Figura 2). Esta funcionalidade conta com a simulação de 4 algoritmos: FCFS (First-Come, First-Served), SJF (Shortest Job First), SRTF (Shortest-Remaining-Time-First) e RR (Round Robin). Dentre estes algoritmos apenas o RR requer outra entrada para a simulação, o quantum de tempo.

Após o usuário iniciar a simulação, o um dos processos vai para a “Execução”, enquanto os outros aparecem na “Fila de Espera” assim que o tempo decorrido da simulação for igual ao tempo de chegada daquele processo. Ao final da simulação, é gerado o “Diagrama de Gantt” desta, permitindo uma melhor análise da simulação.

Algoritmo:

Nº de Processos:

	Tempo de burst	Tempo chegada
P1	<input type="text" value="8"/>	<input type="text" value="0"/>
P2	<input type="text" value="3"/>	<input type="text" value="1"/>
P3	<input type="text" value="5"/>	<input type="text" value="2"/>

Execução

P1

Fila de Espera

P2 P3

Diagrama de Gantt

P1P1P1P1P1P1P1P1P1P2P2P2P3P3P3P3P3

FIGURA 2. Tela de simulação de escalonamento de CPU.

A simulação de Sincronismo de Processo possui simulação para dois algoritmos: Estrita alternância e algoritmo de Peterson. Esta funcionalidade simula o sincronismo entre apenas dois processos. Antes de iniciar a simulação, o usuário deve escolher um algoritmo e informar, para cada processo, se sua Seção Crítica e Seção Não Crítica será rápida ou devagar (veja Figura 3). Ao iniciar a simulação, um processo entra em “Execução”, enquanto o outro fica em “Espera”. Enquanto a simulação estiver sendo executada, dois marcadores indicam em qual seção da execução cada processo está.

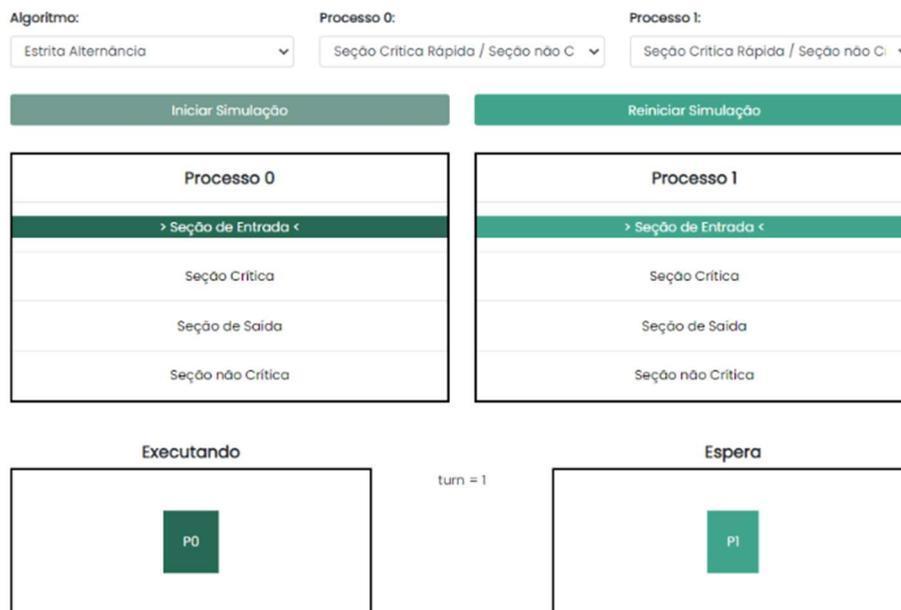


FIGURA 3. Tela de simulação de sincronismo de processos.

A funcionalidade de simulação de deadlock possui dois caminhos distintos durante a execução, dependendo de o usuário marcar ou não a opção “Evitar deadlock”. Caso a opção fique desmarcada, o usuário informar quantas instâncias existem para cada recurso e, quantos recursos cada processo já alocou. Após isso, o usuário pode simular requisições dos processos e visualizar o resultado destas. Caso a opção fique marcada, o usuário deve informar, além do mencionado anteriormente, o máximo de recursos que cada processo pode alocar. Nesse cenário, o sistema utiliza algoritmos para evitar o deadlock, e o usuário pode ainda testar a segurança do sistema e simular requisições dos processos. A Figura 4 demonstra um cenário em que o usuário marcou a opção “Evitar deadlock”.

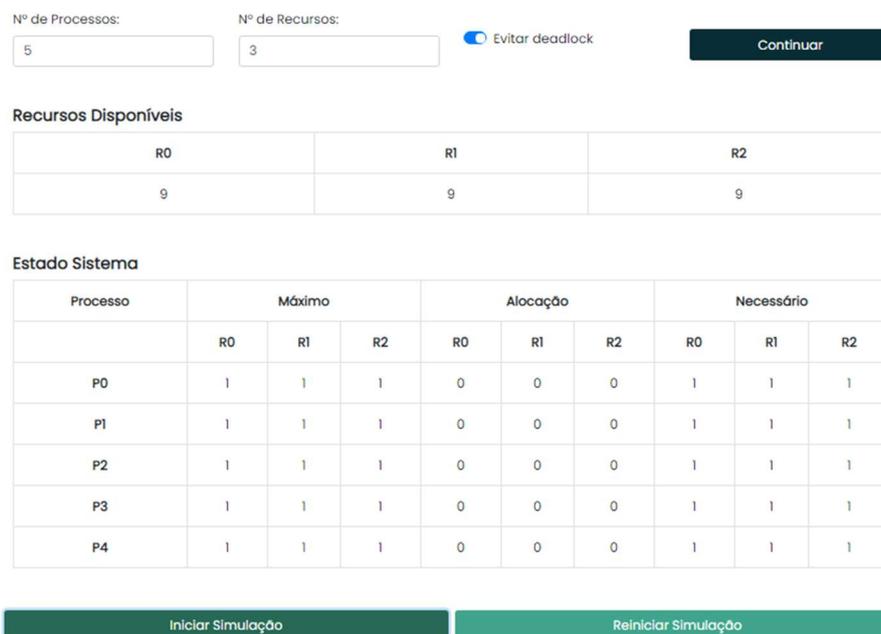


FIGURA 4. Tela de simulação de deadlocks.

A simulação de alocação de memória conta com três algoritmos: First-Fit, Best-Fit e Worst-Fit. Antes de iniciar a simulação, o usuário deve escolher um dos algoritmos e determinar o tamanho da memória (veja Figura 5), que define o número de endereços disponíveis para serem alocados. O usuário pode ainda, gerar processos aleatórios na memória, alocar novos processos, informando seu tamanho, ou remover processos e liberar memória, tendo que informar o número do processo.



Algoritmo: Best-fit

Tamanho da Memória (KB): 10

Criar Memória

Gerar Processos Aleatórios

Memória

P1	P1	P1		P2	P2		P3
----	----	----	--	----	----	--	----

Novo Processo

Tamanho do Processo (KB):

Alocar Processo

Remover Processo

Número do Processo:

Remover Processo

FIGURA 5. Tela de simulação de alocação de memória.

A ferramenta não foi experimentada por alunos, devido ao atraso do calendário escolar no IFSP campus Jacareí, causado pela pandemia. Porém, ainda há a intensão de realizar experimentos com alunos deste campus, enquanto eles cursam a disciplina de sistemas operacionais, a fim de auxiliá-los no processo de aprendizagem desta. Para avaliar a eficiência da ferramenta, pretende-se disponibilizar um questionário para os alunos descreverem como a ferramenta os auxiliou e, através de conversas com o professor que ministrava a disciplina, verificar se o desempenho dos alunos correspondeu às expectativas.

CONCLUSÕES

Tendo em vista a dificuldade existente no aprendizado dos conceitos e algoritmos relacionados a sistemas operacionais, torna-se necessário o desenvolvimento de ferramentas didáticas que auxiliem neste processo. O sistema proposto neste trabalho visa simular o funcionamento de tarefas de um sistema operacional através de animações, utilizadas para facilitar sua compreensão. As simulações permitem entender e observar melhor o funcionamento e as diferenças entre os diversos algoritmos simulados.

O ambiente simulado desenvolvido neste trabalho é voltado para a educação de estudantes da área de tecnologia da informação, e que, por ser web, torna seu acesso mais fácil e atrativo, já que depende apenas de um navegador e internet, coisas corriqueiras para pessoas dessa área. A principal função da ferramenta será auxiliar professores a ensinar o conteúdo da disciplina de Sistemas Operacionais para seus alunos, usando uma ferramenta interativa.

REFERÊNCIAS

- BOS, H.; TANENBAUM, A. S. Sistemas Operacionais Modernos. 4ª edição. São Paulo: Pearson, 2016.
- LOIOLA, R. Geração Y – Revista Galileu – Edição 219. Disponível em: <http://revistagalileu.globo.com/Revista/Galileu/0,,EDG87165-7943-219,00-GERACAO+Y.html>. Acesso em: 18 fev. 2020.
- MADUREIRA, D. Desenvolvimento Web: Entenda Todo o Processo. Disponível em: <<https://usemobile.com.br/desenvolvimento-web-processo/>>. Acesso em: 18 fev. 2020.
- MAZIERO, C. A. Reflexões sobre o ensino prático de Sistemas Operacionais. Anais do X Workshop sobre Educação em Computação (WEI2002), XXII Congresso da SBC, 2002.
- MORAES, E. L. et al. Simulador Web de Algoritmos para Escalonamento de Processos em um Sistema Operacional. Disponível em: <<http://docplayer.com.br/68531303-Simulador-web-de-algoritmos-para-escalonamento-de-processos-em-um-sistema-operacional.html>>. Acesso em: 30 nov. 2019.
- RAMOS, V. Desenvolvimento Web Com Python E Django: Introdução. Disponível em: <<https://pythonacademy.com.br/blog/desenvolvimento-web-com-python-e-django-introducao#hello-world-django>>. Acesso em: 20 jan. 2019
- SCHWABER, K.; SUTHERLAND, J. Guia do Scrum. Um guia definitivo para o Scrum: As regras do jogo. 2017