

Podcast

Disciplina: Frameworks back end

Título do tema: Utilização do Framework Express

Autoria: Leandro C. Cardoso

Leitura crítica: Arthur Gonçalves Ferreira

Abertura:

Olá, ouvinte! No podcast de hoje vamos falar sobre o Node.js

Foi em maio do ano de 2009, que o engenheiro de software dos Estados Unidos, Ryan Dahl o desenvolvedor do Node.js, no momento que estava executando um upload de um site. No qual este site que estava trabalhando, tinha como principal serviço de carregamento de arquivos, foi quando ele teve a inspiração para o desenvolvimento do Node.js, ou simplesmente Node.

Ele percebeu que o site não apresentava qual era o status do carregamento do arquivo, e para saber a evolução do carregamento era necessário a todo o momento executar a consulta no servidor web para certificar o status de carregamento. Vários estudiosos e profissionais da área afirmam que foi a partir dessa situação o surgimento do termo *server-side*, e por esse motivo que o Node.js pode ser classificado um ambiente Javascript para sua execução de modo *server-side*.

Para entender melhor, um exemplo prático de *server-side* é a hospedagem Node.js, desta maneira, são linguagens que são desenvolvidas diretamente pelo servidor, assim a codificação é escrita dentro do servidor. Resumindo, a codificação processa e envia a resposta para o navegador web, diretamente do servidor, ou seja, não “passa” digamos assim, por nenhum intermediário. no qual o navegador web envia a linguagem e executa o processamento por meio do navegador web, ou seja, é o navegador web que se encarrega de processar as informações.

Vários desenvolvedores que utilizam o Node.js, apontam como uns dos diferenciais é em relação ao *single thread*, pois o Node.js realiza o suprimento para que apenas uma thread, possa executar a codificação da aplicação. Assim não é necessária a criação de thread para cada nova requisição. Lembrando que thread trata-se, de certa forma de uma pequena aplicação que tem a função de trabalhar como um subsistema, desta maneira, é uma forma de um processo de autodividir em duas ou mais tarefas, se traduzirmos o termo em inglês para o português seria linha ou encadeamento de execução.

Outro fato importante no Node.js é que nas linguagens de programação consideradas como tradicionais no momento que é executada uma requisição acontece a criação de uma nova thread. Mas no Node.js, dispõe apenas uma thread, que recebe o nome de Event Loop, no qual a sua função é de criação de novos eventos a cada nova requisição, utilizando desta maneira permite que

ocorra diversas operações ao mesmo tempo em áreas diferentes, por esse motivo que recebe o nome quando traduzido para o português de Ciclos de Eventos.

Tendo em vista todas as vantagens o Node.js pode ser utilizado em diferentes locais como para protótipos, devido à sua velocidade e leveza em conexões externas, facilitando a criação de protótipos de APIs com rapidez e agilidade.

Fechamento:

Este foi nosso podcast de hoje! Até a próxima!