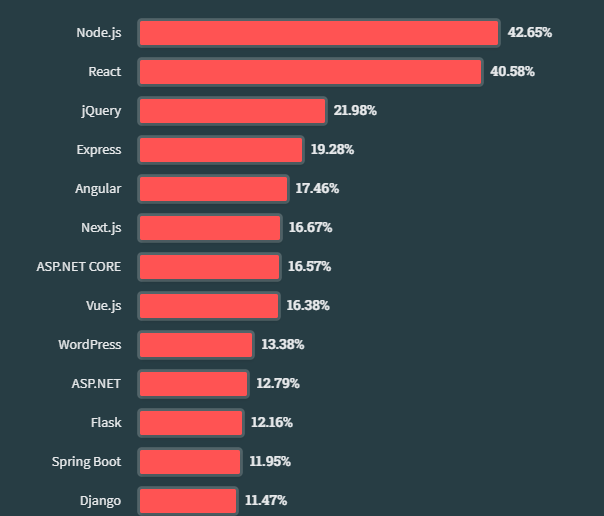
O express também está na lista dos frameworks mais utilizados em 2023 com os mesmos 71.802 votos, ocupando a 4 colocação e sendo o framework node para back-end mais usado atualmente.

O express é uma ferramenta muito poderosa para a construção do lado do servidor, podendo ser usado para construir basicamente qualquer coisa que se envolva com a web, desde gateways para delegar as requisições a micro serviços, servidores de páginas estáticas, API Restful dentre muitas outras abordagens diferentes para usá-lo.

Pontos fortes do express

É um framework minimalista, que visa o desenvolvimento rápido e eficaz, ele tem uma alta compatibilidade com as versões do node, é um dos melhores frameworks no quesito desempenho, fácil e rápido de construir uma api.

comparação com o Hapi

Foi realizado testes de cargas em API simples locais com as seguintes configurações

3000 usuários simultâneos usando a api durante 20 segundos

os dados obtidos foram esses:

**express: máximo de 220 mb usados com um total de 22000 requisições feitas com sucesso , 0 erros.**

**hapi: máximo de 90 mb usados com um total de 0 requisições feitas com sucesso, 10000 erros**

É muito assustador notar que o hapi não conseguiu suprir as necessidades técnicas impostas no teste de carga, esse foi o principal motivo de não termos escolhido o hapi para ser a estrutura do nosso backend, assim o express.

E apesar do hapi ter uma documentação oficial mais completa e mais robusta no quesito performance ele deixou a desejar, e também é uma biblioteca com uma curva de aprendizagem muito alta que pode dificultar o entendimento dos outros integrantes, coisa que o express não é, ele é fácil e prático.

outros pontos que elegeram o express

o express tem uma maneira de lidar com as requisições que chegam pra ele, na qual seria de não bloquear o thread principal, ou seja ele tem como característica um I/O não bloqueante que aumenta em muito se combinado com os recursos de childprocess e worker threads do próprio node, aliás é essa característica de I/O não bloqueante que fez a equipe do angular escolher ele para realizar o'que chamamos de SSR (server side rendering).

ele é simple de ler e entender, o controle que temos também com os middlewares que nós mesmos podemos implementar nos dá um poder muito grande, por exemplo fazer o rastreamento de cada requisição, caso uma requisição falhe ele pega aquela em específico e retorna um erro somente para ela.

Arquitetura arbitrária, pode-se realizar desde um simples MVC(model, view, controller) ou até mesmo em arquitetura em camadas ou hexagonal, ou outras. Tem vasto suporte para pacotes de terceiros que a própria comunidade constrói, como ele é um framework minimalista às vezes ele pode carecer de alguns comportamentos como autenticação via token ou outras formas de autenticação.