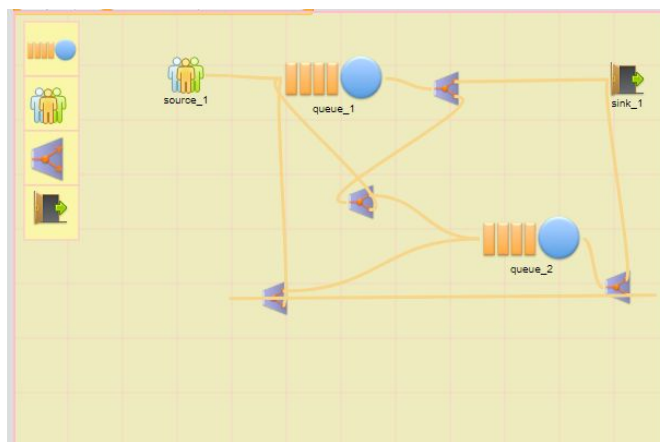




## Relatório

1. O simulador tenta caracterizar uma modelagem de uma simulação de requisições web para um módulo nodeJS, ele recebe uma requisição por vez, logo depois de um tempo aleatório de atendimento ele chaveia entre ir para fila de eventos (para contornar IO) , voltar para execução ou sair do sistema.



- 2.

estados x filas	
execução da requisição	eventos/IO
ocioso	ocioso
ocupado	ocioso
ocupado	ocupado
ocioso	ocupado

3. Supondo que uma requisição A apareça no sistema. Ela será recebida pela pilha de processamento. Será pareada com um programa nodejs Supondo uma requisição A, o processamento vai consumir um pouco do prog1, visto que será executado, e (A,prog1) sairá (A, prog2) Então o que ficará na fila será (A,prog1) Quando A for atendido o processamento vai consumir um pouco do prog1, visto que será executado, e (A,prog1) sairá (A, prog2) Se prog2 for o programa vazio, ele sairá do sistema, pois foi atendido Senão, se prog2 começar com um comando de IO, vai para a fila de IO Senão, se prog2 começar com um comando de cpu, volta para a última posição da fila de processamento na fila de IO, o atendimento será a espera até um evento do nodeJS ocorrer quando isso acontecer, (A, prog2) virará (A, prog3) que poderá ser vazio, encabeçar processamento ou IO e os destinos respectivos

Como no simulador não executaremos de fato um programa, o que acompanhará a requisição será uma sequência tipo [comando, comando, IO, comando, IO, IO, IO]

algo assim. Cada comando terá uma execução aleatória, sendo no caso o tempo de atendimento da primeira fila. Da mesma maneira, com média superior a primeira, será o tempo de cada IO

Suponha que cada processamento tenha tempo médio de 100 microssegundos e cada IO de 25 milissegundos. Então, quando uma requisição chega a ponto de ser atendida, suponha, na fila de processamento, será gerado um evento de saída com média de 100 microssegundos e o comando em seu topo será removido. Ou seja,

[comando, comando,IO] -----> [comando, IO]  
exponencial com média de 100 microssegundos

Depois do atendimento anterior, veja que outro comando ficou no topo do programa, logo essa requisição retornará a fila de processamento. Quando for novamente atendida, terá se tornado [IO]. Irá então, para fila de IO.

Esperará o tempo de fila e o tempo de processamento(no caso com média de 25 milissegundos) para ficar com [] ou seja, o programa acabou e ele é removido do sistema.

.