Gabriel Campos Albino
Paulo Arthur Sens Coelho
Erik Kazuo Sugawara

Relatório

Florianópolis, SC

2022

Gabriel Campos Albino
Paulo Arthur Sens Coelho
Erik Kazuo Sugawara

Relatório

Trabalho apresentado para avaliação de uma implementação de uma biblioteca para comunicação confiável e algoritmo distribuído, na disciplina INE 5418 - Computação Distribuída, sob Orientação do Prof. Dr. Odorico Machado Mendizabal

Universidade Federal de Santa Catarina
Departamento de Informática e Estatística
Programa de Graduação

Orientador: Odorico Machado Mendizabal

Florianópolis, SC
2022

Gabriel Campos Albino
Paulo Arthur Sens Coelho
Erik Kazuo Sugawara
Relatório/ Gabriel Campos Albino
Paulo Arthur Sens Coelho
Erik Kazuo Sugawara. – Florianópolis, SC, 2022-
13 p. : il. (algumas color.) ; 30 cm.

Orientador: Odorico Machado Mendizabal

Relatório (Graduação) – Universidade Federal de Santa Catarina
Departamento de Informática e Estatística
Programa de Graduação, 2022.

1. Computação Distribuída. 2. Sockets. 3. Ordem Causal. 4. Ordem Total. 5. Biblioteca.

CDU 02:141:005.7

Gabriel Campos Albino
Paulo Arthur Sens Coelho
Erik Kazuo Sugawara

Relatório

Trabalho apresentado para avaliação de uma implementação de uma biblioteca para comunicação confiável e algoritmo distribuído, na disciplina INE 5418 - Computação Distribuída, sob Orientação do Prof. Dr. Odorico Machado Mendizabal

---

Odorico Machado Mendizabal
Orientador

Florianópolis, SC
2022

Lista de ilustrações

Lista de tabelas

Sumário

Identificação

Universidade Federal de Santa Catarina
Departamento de Informática e Estatística
Ciências da Computação
INE 5425 - Modelagem e Simulação

Contato

Gabriel Campos Albino
Paulo Arthur Sens Coelho
Erik Kazuo Sugawara
Telefone: +55 (48) 99814-7971
Email: erik.sugawara@grad.ufsc.br

Parte I

Biblioteca de Comunicação Confiável

1 Sobre o Trabalho

1.1 Introdução

Um sistema distribuído é caracterizado por componentes localizados em uma rede de computadores, que se comunicam e coordenam suas ações através de troca de mensagens e que podem sofrer influências internas ou externas, as quais podem prejudicar a sua interoperabilidade. Através da concorrência entre os componentes comunicantes, o estado de seus dados se tornam inconsistentes a medida que não existe uma sincronização. Seja por conta da ausência de um relógio global ou de uma memória compartilhada, se torna impossível realizar uma sincronização perfeita entre as trocas de mensagens. Além disso, em seu aspecto físico, componentes são suscetíveis a falha e podem impedir o funcionamento do sistema. Sendo assim, o objetivo deste trabalho é criar uma biblioteca de comunicação confiável que seja capaz de garantir a ordem no recebimento e envio de mensagens seguindo critérios de ordem causal e total, de modo que os participantes da comunicação consigam estabelecer troca de mensagens do tipo 1 : 1 e 1 : n.

1.2 Metodologia

Para realizar a criação da biblioteca foi escolhida a linguagem Python por sua versatilidade e legibilidade. Em conjunto, foi usado a biblioteca de Multiprocessos para permitir que os processos que instanciam os sockets sejam possíveis de serem executados em paralelo, garantindo o recebimento e envio de mensagens de múltiplos nodos. Cada nodo contém dois processos que instanciam sockets: o primeiro é utilizado para o recebimento de mensagens e o segundo é utilizado para enviar mensagens, uma vez que não conseguimos reutilizar o mesmo socket para realizar operações de envio e recebimento. Para garantir a entrega dos dados, foi utilizado o Protocolo TCP.

1.2.1 Protocolo TCP

Foi utilizado o domínio AF_INET para criarmos um socket capaz utilizar os endereços com IP e Porta. Além disso, a garantia do recebimento de pacotes facilita a elaboração da biblioteca.

Explicar mais sobre o tcp

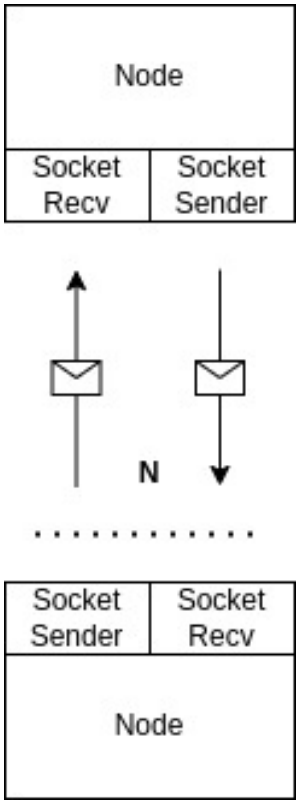

Figura 1 – Estrutura de um nodo

1.2.2 Nodo

Cada nodo contem dois sockets, um para recebimento de mensagens e outro para envios. Contem o enderecos de outros nodos e seu id.

Explicar os atributos etc.

1.2.3 Ordem Causal

1..1 Explicar a ordem causal

1.2.4 Ordem Total

1..n Explicar a ordem total

2 Desenvolvimento

2.1 Biblioteca

2.2 Execução

2.3 Resultados

2.4 Algoritmo

Ring

2.5 Conclusão

2.6 Referências

ANEXO A – Biblioteca

A.1 Código

ANEXO B – Algoritmo