



COMUNICAÇÃO DE DADOS E REDES DE COMPUTADORES

TERMO DE ABERTURA DE PROJETO - TAP

REPOSITÓRIO DO PROJETO:

https://github.com/joaolucassilvadev/sistema_distribuidos

IDENTIFICAÇÃO DO PROJETO

NOME DO PROJETO:

Chat Distribuído em Tempo Real

GERENTE:

Pedro Arthur Da Silva Guimarães

DISCIPLINA:

Sistemas Distribuídos

PARTICIPANTES

ARTHUR SAMPAIO PEREIRA

CARLOS RENAN ANDRADE PEREIRA

JOAO LUCAS SILVA DA SILVA

PAULO VINICIUS ARAUJO SILVA

PEDRO ARTHUR DA SILVA GUIMARÃES

HISTÓRICO DE REGISTRO

1.0	18/04/2025	Início de Elaboração do documento.
1.1	19/04/2025	Consolidação do tema e divisão de atividades entre os integrantes.
1.2	21/04/2025	Alinhamento dos últimos detalhes para conclusão da elaboração do documento

1. VISÃO GERAL DO PROJETO

Este projeto tem como objetivo o desenvolvimento de um sistema de chat em tempo real baseado em uma arquitetura distribuída peer-to-peer (P2P), utilizando a plataforma InterSCity para descoberta e integração entre os nós. A proposta visa eliminar a necessidade de um servidor central, permitindo que os usuários se comuniquem diretamente entre si de forma segura, escalável e tolerante a falhas. O sistema permitirá a troca de mensagens em tempo real, gerenciamento de contatos, armazenamento de mensagens offline e visualização da rede por meio de um dashboard

2. JUSTIFICATIVA

Com o avanço das tecnologias de redes distribuídas e a crescente demanda por sistemas descentralizados, surge a oportunidade de explorar arquiteturas peer-to-peer (P2P) aplicadas à comunicação em tempo real. Este projeto propõe o desenvolvimento de um sistema de chat distribuído utilizando a plataforma InterSCity, eliminando a dependência de servidores centrais e promovendo a comunicação direta entre usuários. Essa abordagem possibilita uma maior resiliência, escalabilidade e descentralização.

3. REQUISITOS

- Acesso aos dados dos sensores instalados nos contêineres de lixo da cidade através das APIs da Plataforma InterSCity.
- Desenvolvimento de algoritmos para análise de dados e identificação de contêineres prioritários para coleta.
- Implementação de um sistema de otimização de rotas para os caminhões de coleta, considerando a localização dos contêineres e a capacidade dos veículos.
- Interface de usuário amigável para os funcionários responsáveis pela coleta de resíduos e para os gestores municipais acompanharem o desempenho do sistema.

4. OBJETIVOS

Objetivo Geral:

Desenvolver um sistema de chat distribuído em tempo real baseado em comunicação peer-to-peer, com descoberta e comunicação entre nós utilizando a plataforma InterSCity.

Objetivos Específicos:

- Projetar uma arquitetura P2P utilizando InterSCity para descoberta e troca de mensagens entre os nós;
- Implementar funcionalidades básicas de um sistema de chat (envio/recebimento de mensagens, contatos conectados, histórico local);
- Permitir comunicação assíncrona, com armazenamento temporário de mensagens para nós offline;
- Simular e testar o sistema em múltiplos nós distribuídos;
- Avaliar a tolerância a falhas e o desempenho do sistema em diferentes cenários de carga;
- Documentar todas as etapas de desenvolvimento conforme padrões técnicos.

4. ESCOPO

Escopo

O escopo deste projeto contempla o desenvolvimento de um sistema de comunicação em tempo real baseado em uma arquitetura distribuída peer-to-peer (P2P), utilizando a plataforma InterSCity como suporte à descoberta de serviços e integração dos nós participantes. O sistema será estruturado para promover a descentralização da comunicação, a resiliência a falhas e a escalabilidade horizontal.

Inclui:

Implementação de clientes (usuários) como nós distribuídos:

Cada usuário do sistema será representado por um nó autônomo capaz de estabelecer comunicação direta com outros nós da rede. Esses clientes atuarão de forma descentralizada, sem a necessidade de um servidor central, refletindo a proposta de um sistema verdadeiramente distribuído.

Descoberta de nós e conexão P2P via InterSCity:

A descoberta de outros usuários ativos na rede será realizada através da integração com a plataforma InterSCity. Isso permitirá que os nós localizem e se conectem dinamicamente a outros participantes, facilitando a formação e manutenção da rede de comunicação.

Troca de mensagens em tempo real utilizando WebSockets ou tecnologia similar:

As mensagens serão transmitidas de forma assíncrona e em tempo real por meio do protocolo WebSocket ou alternativas compatíveis, assegurando baixa latência na

comunicação e suporte a múltiplas conexões simultâneas entre pares.

Fila de mensagens offline:

Para garantir a continuidade e integridade das conversas mesmo em cenários de indisponibilidade temporária de algum nó, será implementado um sistema de fila local para armazenamento de mensagens pendentes. Assim, as mensagens poderão ser entregues assim que o destinatário voltar a ficar online.

Dashboard simples (CLI ou Web) para visualização da rede e mensagens:

Será desenvolvido um painel de controle, podendo ser em interface de linha de comando (CLI) ou via interface web, com o objetivo de oferecer aos usuários uma visualização clara da rede de contatos disponíveis, mensagens recebidas, histórico de conversas e status de conexão dos nós.

Documentação técnica e relatório final do projeto:

Todas as etapas do projeto, desde a concepção até a implementação e testes, serão devidamente documentadas. Isso inclui a descrição da arquitetura adotada, tecnologias utilizadas, decisões de projeto, resultados dos testes de desempenho e instruções de uso. O relatório final seguirá os padrões técnicos acadêmicos exigidos, consolidando a entrega formal do trabalho.

5. RECURSOS

Recursos Necessários

Para a execução do projeto, serão empregados diversos recursos que abrangem componentes de software, hardware e recursos humanos, essenciais para o desenvolvimento, simulação, testes e validação do sistema proposto.

Software (SW)

Python ou Node.js: Linguagens de programação utilizadas para o desenvolvimento dos clientes (nós) e da lógica de comunicação entre pares. Ambas oferecem suporte robusto para redes, WebSockets e integração com APIs REST.

InterSCity: Plataforma de middleware urbano inteligente utilizada para facilitar a comunicação entre os nós distribuídos, fornecendo suporte à descoberta de serviços e troca de dados entre elementos do sistema.

WebSockets: Tecnologia de comunicação bidirecional em tempo real, utilizada para envio e recebimento de mensagens entre os nós do sistema.

Git: Sistema de controle de versão distribuído, que será empregado para gerenciar o código-fonte, possibilitar colaboração entre os desenvolvedores e garantir rastreabilidade das alterações.

Docker: Plataforma de virtualização leve utilizada para empacotar e distribuir os componentes da aplicação em containers, garantindo portabilidade, facilidade de configuração e escalabilidade dos serviços.

Hardware (HW)

Computadores pessoais: Equipamentos dos integrantes da equipe que servirão como ambiente de desenvolvimento, execução dos nós simulados e testes do sistema.

Conexão à internet: Recurso essencial para garantir a comunicação entre os nós distribuídos, que estarão em diferentes locais físicos durante a simulação e testes do sistema.

Servidor de testes (opcional): Máquina com recursos dedicados para hospedar uma instância de teste do sistema, centralizar logs, realizar testes de carga e analisar o desempenho em cenários mais realistas de operação.

Recursos Humanos (RH)

4 estudantes/programadores: Responsáveis pelo planejamento, desenvolvimento, testes e documentação do sistema. Cada membro poderá assumir funções específicas como: desenvolvimento backend, integração com InterSCity, testes e visualização.

Professor/orientador: Profissional responsável por acompanhar o progresso do projeto, fornecer orientações técnicas e teóricas, além de garantir a aderência aos objetivos acadêmicos e práticos da proposta.

Usuários para testes: Grupo de usuários (podendo incluir outros alunos da instituição) que serão convidados para utilizar o sistema em ambiente controlado, com o intuito de validar a usabilidade, desempenho e robustez da aplicação.

6. CRONOGRAMA DE IMPLANTAÇÃO

Fase	Período Planejado	Status	Relatório de Tarefas
Planejamento e Preparação	<ul style="list-style-type: none">18/04/25 - 23/04/25	Concluída	Todas as atividades de definição de requisitos, levantamento de recursos e design inicial da arquitetura foram finalizadas com sucesso dentro do prazo estipulado.
Desenvolvimento e Testes Iniciais	<ul style="list-style-type: none">24/04/25 - 02/06/25	Concluída	A implementação da interface básica do chat e a troca de mensagens em tempo real foram concluídas. A integração inicial com a InterSCity e os testes de componentes isolados também foram finalizados.
Integração e Testes do Sistema	<ul style="list-style-type: none">03/06/25 - 24/06/25	Atrasada	A implantação da versão piloto foi concluída, mas os testes de simulação de falhas e mensagens offline se estenderam por uma semana além do prazo, finalizando em 30/06. A coleta de feedback está sendo iniciada com atraso.

Implantação Completa	<ul style="list-style-type: none"> 25/06/25 - 07/07/2 	Em Andamento	O desenvolvimento do dashboard foi iniciado. A implementação do sistema funcional completo está em progresso, porém impactada pelo atraso da fase anterior.
Avaliação e Ajustes Finais	<ul style="list-style-type: none"> 08/07/25 - 16/07/25 	Não Iniciada	As atividades desta fase ainda não foram iniciadas.

Análise de Desvios e Riscos

Atraso Identificado: A fase de "Integração e Testes do Sistema" sofreu um atraso de uma semana. A complexidade na simulação de cenários de falhas e na garantia da entrega de mensagens offline demandou mais tempo de desenvolvimento e depuração do que o previsto.

Impacto: Este atraso está impactando o início das atividades da fase de "Implantação Completa".

Ação Corretiva: A equipe irá realocar esforços para acelerar o desenvolvimento do dashboard e da documentação técnica para mitigar o impacto no prazo final de entrega do projeto.

7. Cronograma de GANTT Atualizado

Fase	Atividades	Início Planejado	Fim Planejado	Status em 30/06/2025
Planejamento e Preparação	<ul style="list-style-type: none"> Definição de requisitos e arquitetura 	18/04/25	23/04/25	Concluída
Desenvolvimento e Testes Iniciais	<ul style="list-style-type: none"> Implementação da interface básica do chat (cliente) Desenvolvimento da troca de mensagens em tempo real Integração inicial com a plataforma InterSCity Testes dos componentes isolados 	24/04/25	02/06/25	Concluída
Integração e Testes do Sistema	<ul style="list-style-type: none"> Implantação de uma versão piloto entre múltiplos nós Testes com simulação de falhas, mensagens offline e reconexão Coleta de feedback e ajustes no sistema 	03/06/25	24/06/25	Concluída com Atraso
Implantação Completa	<ul style="list-style-type: none"> Implantação do sistema funcional completo com dashboard e histórico Execução de testes finais e validação dos critérios do projeto 	25/06/25	07/07/25	Em Andamento

	<ul style="list-style-type: none"> • Documentação técnica e entrega final 			
Avaliação e Ajustes Finais	<ul style="list-style-type: none"> • Revisão geral do sistema com base em testes finais • Ajustes finais de usabilidade e performance • Finalização da documentação e preparação da entrega 	08/07/25 – 16/07/25		Pendente

8. GRÁFICO DE GANTT

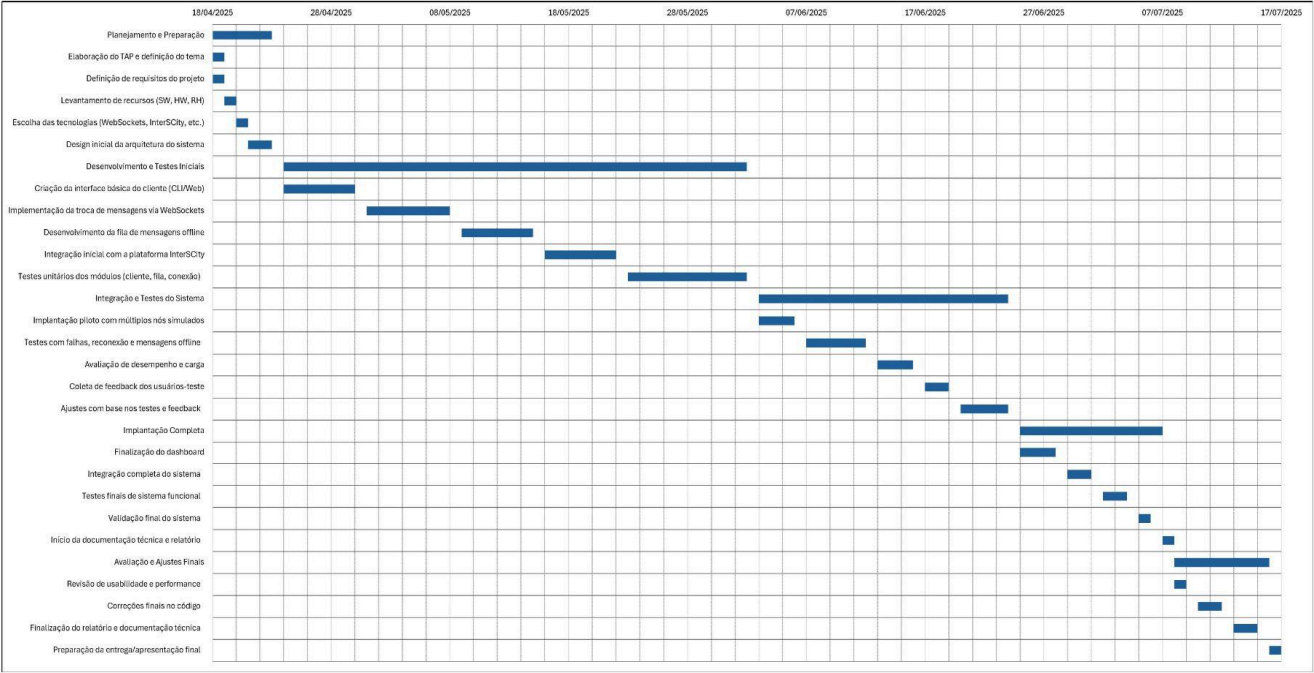


Figura 1: Gráfico de Gantt do projeto