

Etapa 1 - Concepção e Arquitetura Inicial

Nome:

Hanny Emilly Rodrigues Sousa

Matrícula:

32211BSI020

Título do Projeto:

Sistema Distribuído de Entregas com Drones

Descrição do Tema

O projeto simula um sistema distribuído para entregas urbanas usando drones autônomos. A cidade é representada por uma malha de coordenadas, e os pedidos de entrega são gerenciados por um servidor central.

Drones autônomos se comunicam com o servidor e com pontos de entrega, movimentando-se pelo mapa, buscando pacotes e realizando entregas.

Entidades principais:

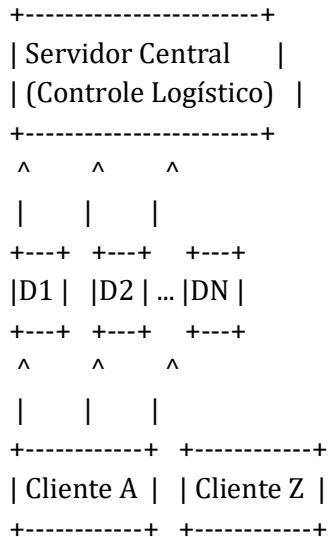
- Servidor Central (Controle Logístico)
- Drones (Agentes Autônomos)
- Clientes/Pontos de Entrega
- Mapa da Cidade (Recurso Compartilhado)

Metas de Sistemas Distribuídos

- Escalabilidade: adicionar mais drones ao sistema.
- Tolerância a Falhas: realocação de tarefas caso um drone falhe.
- Disponibilidade: sistema continua mesmo com falhas de drones.
- Compartilhamento de Recursos: uso do mapa e tarefas.
- Sincronização: evitar colisões entre drones.
- Exclusão Mútua: controle de acesso a pacotes/mapa.
- Comunicação Cliente-Servidor.

- Eleição de Líder (futuramente).

Diagrama da Arquitetura Inicial (Texto)



Todos acessam o Mapa da Cidade (recurso compartilhado)

Descrição de Cada Componente no Diagrama

Servidor Central:

- Responsabilidades: Gerenciar pedidos, monitorar drones, alocar entregas, manter o estado do mapa.
- Tipo de Comunicação: Cliente-servidor (TCP) com clientes e drones.
- Processo ou Thread: Processo independente.

Drone (Agente Autônomo):

- Responsabilidades: Receber tarefas, se movimentar, entregar pacotes, reportar status.
- Tipo de Comunicação: Cliente-servidor com servidor e ponto-a-ponto com outros drones (opcional).
- Processo ou Thread: Thread ou processo autônomo.

Cliente / Ponto de Entrega:

- Responsabilidades: Gerar pedidos, enviar ao servidor, receber notificações.
- Tipo de Comunicação: Cliente-servidor (TCP) com o servidor.
- Processo ou Thread: Processo simples.

Mapa da Cidade (Recurso Compartilhado):

- Responsabilidades: Representar posições dos drones, pacotes e destinos.
- Tipo de Comunicação: Acesso controlado via servidor.
- Processo ou Thread: Não é processo — é uma estrutura de dados compartilhada.

Resumo das Próximas Etapas

- Threads/Processos: Drones como threads ou processos concorrentes.
- Sockets: Comunicação via TCP entre drones, servidor e clientes.
- RPC: Chamadas como `alocar_entrega(drone)` usando RPC.
- Mensagens: Fila de pedidos, status de drones.
- Sincronização: Consistência do mapa e tarefas.
- Exclusão Mútua: Garantir que drones não colidam ou peguem o mesmo pedido.
- Eleição de Líder: Se o servidor falhar.
- Tolerância a Falhas: Redundância e reatribuição de tarefas.
- Segurança: Autenticação e proteção de conexões.