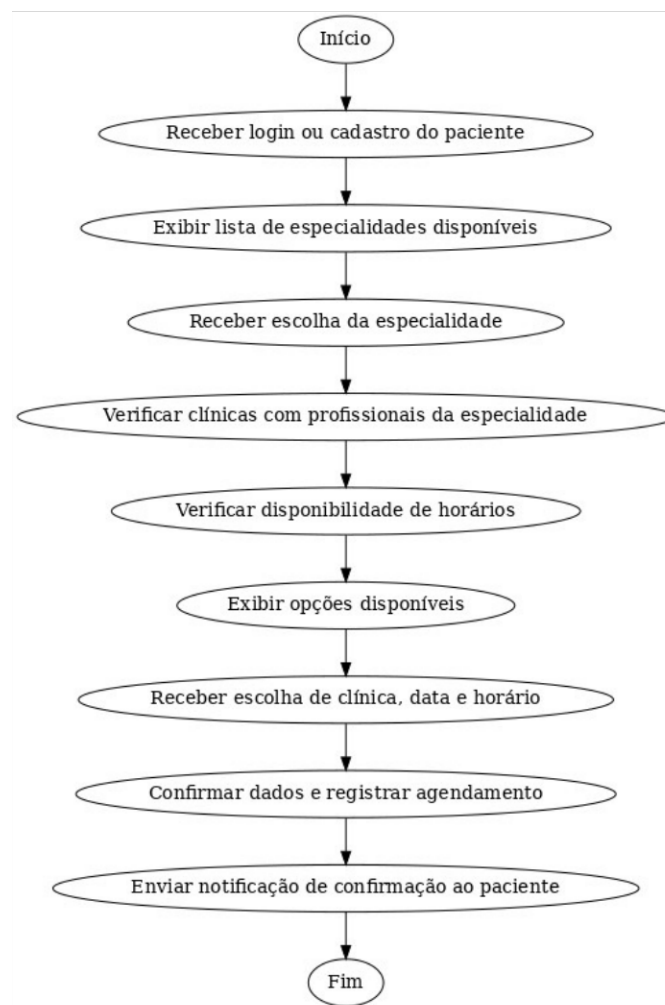


Parte 1 – Criação do Fluxograma

Sistema escolhido: Sistema de agendamento de consultas em uma rede de clínicas.

O sistema permite que usuários realizem agendamentos de consultas médicas em clínicas interligadas digitalmente. O processo envolve o cadastro do paciente, escolha da especialidade, verificação de disponibilidade nas unidades da rede, seleção da data e hora, confirmação do agendamento e envio de notificações ao paciente.

1. Etapas principais do processo (ponto de vista do sistema):
2. Início do processo.
3. Receber login ou cadastro do paciente.
4. Exibir lista de especialidades disponíveis.
5. Receber escolha da especialidade.
6. Verificar clínicas com profissionais da especialidade escolhida.
7. Verificar disponibilidade de horários.
8. Exibir opções disponíveis.
9. Receber escolha da clínica, data e horário.
10. Confirmar dados e registrar agendamento.
11. Enviar notificação de confirmação ao paciente (e-mail/SMS).
12. Fim do processo.



Parte 2 – Reflexão para Transição ao Distribuído

Ao analisar o fluxograma, é possível identificar diversos componentes do sistema que poderiam ser estruturados como serviços independentes (microserviços) em uma arquitetura distribuída:

- Serviço de Autenticação: responsável pelo login e cadastro de usuários, incluindo validação de credenciais e geração de tokens de acesso.
- Serviço de Catálogo Médico: gerencia a lista de especialidades médicas e os profissionais associados a cada clínica.
- Serviço de Disponibilidade: consulta a agenda dos profissionais em tempo real para verificar horários livres.
- Serviço de Agendamento: processa a escolha do usuário e realiza o registro definitivo do agendamento no sistema.
- Serviço de Notificação: envia confirmações e lembretes de consulta via e-mail ou SMS.

Esses serviços poderiam ser distribuídos entre servidores em diferentes regiões para garantir disponibilidade, balanceamento de carga e tolerância a falhas. A comunicação entre microserviços seria feita por meio de APIs REST ou mensageria assíncrona, utilizando protocolos como HTTP/gRPC ou filas como RabbitMQ. A infraestrutura poderia ser orquestrada via Kubernetes e hospedada em nuvem, com replicação de dados entre regiões e autenticação centralizada.