1. **Início do Processo (Start)**: Tudo começa quando um novo agendamento é solicitado.
2. **Paciente Agenda (Lambda: Invoke PacienteAgenda)**: Essa é a primeira ação. A função PacienteAgenda é ativada para registrar a solicitação do paciente.
3. **Sistema Verifica Disponibilidade (Lambda: Invoke SistemaVerificaDisp)**: Depois, o sistema checa se há horários ou recursos disponíveis para o atendimento.
4. **Cliente Espera (Wait state ClienteEspera)**: Aqui, o processo entra em um estado de espera. Isso é útil para, por exemplo, aguardar um tempo antes de continuar, como dar um prazo para o cliente confirmar.
5. **Disponibilidade:**
   * **Disponível (Succeed state Disponível)**: Se o resultado for positivo, o processo segue por este caminho e o estado é concluído com sucesso.
   * **Indisponível (Fail state Indisponível)**: Se o resultado for negativo, o processo falha por este caminho.
6. **Sistema Registra (Lambda: Invoke SistemaRegistra)**: Após a verificação de disponibilidade, o sistema registra o agendamento de fato.
7. **Paciente Consulta (Lambda: Invoke PacienteConsulta)**: Agora é a hora da consulta. O processo invoca a função que gerencia o atendimento médico em si.
8. **Avaliação:**
   * **Paciente Liberado (Succeed state PacienteLiberado)**: Se a avaliação for de alta, o paciente é liberado, e o processo segue com sucesso por este caminho.
   * **Paciente Interna (Fail state PacienteInterna)**: Se for necessário internamento ou um tratamento mais complexo, o processo falha neste caminho para indicar que uma ação adicional é necessária (uma internação).
9. **Paciente Recebe Alta (Lambda: Invoke PacienteRecAlta)**: Após ser liberado, o paciente recebe um documento ou notificação de alta.
10. **Finaliza Protocolo (Lambda: Invoke FinalizaProtocolo)**: Por fim, a última etapa é executada para encerrar o protocolo de atendimento, arquivando todas as informações.
11. **Fim do Processo (End)**: O fluxo de trabalho é concluído.