

## 1. Contexto

- **Objetivo do Sistema:** O sistema é projetado para simular e visualizar processos de cálculo pi, facilitando o entendimento e a análise de sistemas concorrentes e processos de comunicação.
- **Usuários Finais:** Estudantes, professores e pesquisadores de ciência da computação interessados em cálculo pi.
- **Sistemas Externos:** O sistema pode interagir com APIs de terceiros para carregar dados ou código (por exemplo, gists do GitHub para carregar programas de cálculo pi).

## 2. Containers

- **Interface Web:** Interface do usuário baseada em navegador que permite aos usuários interagir com a simulação visual do cálculo pi.
- **Motor de Simulação (PiCalcSimulation):** Componente back-end responsável pela lógica de simulação dos processos de cálculo pi, usando d3.js para a visualização.
- **Servidor Web:** Servidor que hospeda a aplicação web e gerencia as interações entre a interface do usuário e o motor de simulação.

## 3. Componentes

- **Manipulador de Simulação:** Gerencia o ciclo de vida das simulações de cálculo pi, incluindo inicialização, pausa, reprodução e reset.
- **Gerenciador de Processos (Config):** Mantém o estado dos processos, incluindo definições, nós e processos em execução, e gerencia a criação e o término dos processos.
- **Visualizador de Dados (d3.js):** Responsável pela representação gráfica dos processos, canais e ações do cálculo pi.
- **Controlador de Entrada:** Gerencia a entrada do usuário, como código de cálculo pi e configurações de simulação, e passa essas entradas para o motor de simulação.

## 4. Código

- **Classes do Cálculo Pi:** Inclui classes como Channel, ProcCall, Action, Receive, Send, Tau, Alt, que representam diferentes elementos e ações no cálculo pi.
- **Funções de Utilidade:** Inclui funções como portToPicalc, que ajudam na conversão de dados para o formato necessário para simulação e visualização.