## 6.5 Projeto do Sistema Distribuído

Sistema distribuído é aquele em que os componentes localizados em um computador estão conectados uns aos outros em uma rede, se comunicam e coordenam suas ações apenas passando mensagens. Essa definição leva as seguintes características de sistemas distribuídos: simultaneidade de componentes, falta de Relógio global e falhas de componentes individuais. Dollimore e Kindberg (2007)

### **6.5.1 Procedimentos para Tratamentos dos Desafios**

Esta seção descreve os principais requisitos necessários para um sistema distribuído apresentar interoperabilidade funcional. Esses requisitos são: heterogeneidade, escalabilidade, abertura, segurança, manuseio de falhas, concorrência e transparência.

6.5.1.1 Heterogeneidade

Com a internet, é possível aos usuários acessarem serviços e executarem aplicativos por meio de um conjunto heterogêneo de computadores e redes. Dollimore e Kindberg (2007). A heterogeneidade se aplica aos seguintes aspectos:

1. redes: através de protocolos de internet é possível ser realizada a comunicação;
2. hardwares de computador: diferentes padrões de placas de rede fazem que aconteça diferentes implementações. Exemplo: placa de rede Ethernet têm uma implementação diferente daqueles que possuem placas de outros padrões, isto acontece também com arquiteturas de microprocessadores (Intel, ARM, etc.);
3. sistemas operacionais: diferentes sistemas operacionais trazem consigo diferentes formas de programação e interface para protocolos de internet;
4. linguagens de programação: cada linguagem de programação possui suas singularidades, seja no tratamento de vetores, registros e variáveis. É necessário tratar essas diferenças para que não haja erros de informações e haja uma comunicação efetiva;
5. implementação de diferentes desenvolvedores: é necessário que diferentes desenvolvedores utilizem de padrões comuns no desenvolvimento para que haja comunicação entre os diferentes sistemas.

O projeto Deu Pet é uma solução distribuída. A aplicação *back-end* é implementada utilizando a linguagem de programação Java. O *back-end* e também o *front-end* da aplicação, serão executadas por meio do *container* thymeleaf.

Também é utilizada neste sistema a arquitetura *Representational State Transfer* (REST), para acesso aos serviços providos pela aplicação servidora, linguagem Java e estilo de programação SOA/REST. O sistema utiliza o protocolo *Hypertext Transfer Protocol Secure* (HTTPS) para comunicação de dados entre as aplicações.

6.5.1.2 Escalabilidade