

Trabalho Prático

Protocolo de comunicação para aplicações *smart*

Prof. Kalinka Regina Lucas Jaquie Castelo Branco
Estag. PAE: Mariana Rodrigues

Informações gerais

Neste trabalho, o grupo de alunos (04 alunos no máximo) deverá desenvolver e implementar um protocolo de camada de aplicação para **uma** das aplicações propostas, encontradas no final da especificação deste trabalho:

- **Aplicação 01:** Estufa inteligente
- **Aplicação 02:** Incubadora Inteligente
- **Aplicação 03:** Sala de aula Inteligente

O grupo deverá seguir os requisitos apresentados para a aplicação escolhida. O *core* do protocolo consiste nas mensagens trocadas para o funcionamento de cada aplicação.

O trabalho possui duas etapas: desenvolvimento do protocolo e implementação prática.

Etapa de desenvolvimento

No desenvolvimento, o grupo deverá criar mensagens que serão trocadas na camada de aplicação pelos componentes do sistema escolhido de forma que os requisitos de funcionamento sejam atendidos.

Cada mensagem deverá ser documentada separadamente, explicitando a sua estrutura (mensagens com a mesma função como por exemplo, mensagens de dados de sensores podem ser documentadas juntas). Por exemplo, um protocolo para troca de e-mails entre dois usuários pode conter uma mensagem chamada `ENVIAR_EMAIL`, com os campos:

1. e-mail do remetente (obrigatório)
2. número de destinatários (obrigatório)
3. e-mail(s) do(s) destinatário(s) 1...n (obrigatório)
4. corpo da mensagem (opcional)
5. prioridade (opcional)

A forma como a troca de mensagens entre os componentes atende aos requisitos de funcionamento do sistema poderá ser descrita por meio de texto ou imagens (por exemplo, diagrama de sequência). É válido frisar que nem todo requisito de funcionamento será atendido por uma mensagem; é possível que o requisito seja atendido por um campo de uma mensagem específica.

O protocolo deverá possuir um *header* específico para identificá-lo. O grupo pode consultar *headers* de outros protocolos da camada de aplicação (HTTP, MQTT, SMTP, etc) para inspiração.

Etapa de implementação

Nesta etapa, o grupo deverá implementar o protocolo desenvolvido na etapa anterior utilizando *sockets*, que serão comentados e apresentados na camada de transporte. Nesta etapa, as seguintes regras se aplicam:

- Não há restrições quanto à linguagem de programação utilizada; várias delas (como C/C++, Java e Python) têm interface de *sockets* já implementadas em bibliotecas.
- Cada componente do sistema deverá ser um **processo** separado no sistema operacional (SO). A implementação de cada processo é livre, podendo ter mais de uma *thread*.
- O grupo deverá explicitar o SO e o compilador utilizados para que o trabalho possa ser devidamente avaliado.
- O software deverá explicitar a troca de mensagens (imprimindo na tela, por exemplo) e implementar as situações que mostram os diversos requisitos de funcionamento.

Avaliação

O trabalho dos grupos será avaliado segundo os seguintes critérios:

- Cumprimento dos requisitos da aplicação (3,0 pts)
- Descrição do protocolo desenvolvido (2,0 pts)
- Funcionamento da aplicação seguindo as regras estabelecidas (3,0 pts)
- Clareza da implementação (documentação do código) (2,0 pts)

Prazos

Entrega 1: 26/05/2019 23h59

Cada grupo deverá enviar pelo Moodle da disciplina um relatório contendo:

- Integrantes do grupo
- Aplicação escolhida
- Descrição do protocolo desenvolvido, com descrição do *header*, mensagens e como os requisitos funcionais são cumpridos.

Observação: O relatório não precisa ser rebuscado! Sejam claros e assertivos na descrição do protocolo.

Entrega 2: 23/06/2019 23h59

O grupo deverá enviar pelo Moodle da disciplina o código fonte da implementação do protocolo em *sockets*, juntamente com as informações de SO/compilador utilizados para avaliação.

Caso o envio pela plataforma apresente problemas, a entrega poderá ser feita por e-mail. O código deve ser enviado para a estagiária PAE (rodrigues.mariana@usp.br), com cópia para a professora (kalinka@icmc.usp.br), **respeitando os prazos de entrega**.

Observação: Façam um código bem documentado e usem nomes bem descritivos nas variáveis. Grupos que entregarem códigos mal documentados ou de difícil compreensão serão convocados para uma arguição oral, além de perderem pontos no critério **Clareza da implementação**.

Dúvidas

Dúvidas a respeito do trabalho podem ser tratadas com a estagiária PAE pelo e-mail rodrigues.mariana@usp.br.

Aplicação 01: Estufa Inteligente

Estufas são utilizadas no cultivo de plantas em condições controladas de temperatura, umidade, luminosidade, nível de CO₂, entre outras variáveis.

Nesta aplicação, as condições da estufa são configuradas, monitoradas e controladas por um gerenciador, que se comunica com os sensores/atuadores dentro da estufa e pode receber configurações ou responder consultas de um cliente externo.

Componentes

- Sensores:
 - Temperatura interna
 - Umidade do solo
 - Nível de CO₂
- Atuadores:
 - Aquecedor (quando ligado, aumenta a temperatura)
 - Resfriador (quando ligado, diminui a temperatura)
 - Sistema de irrigação (quando ligado, aumenta a umidade do solo)
 - Injetor de CO₂ (quando ligado, aumenta a concentração do gás)
- Gerenciador, que funciona como o servidor da aplicação.
- Cliente, que pode configurar os parâmetros da estufa e requisitar valores das leituras dos sensores.

Princípio de operação

O Gerenciador deve manter as leituras dos sensores entre os valores máximo e mínimo configurados

Requisitos Funcionais

1. Sensoriamento:
 - 1.1. Todos os sensores tem um identificador único.
 - 1.2. Os sensores devem se conectar ao Gerenciador e se identificar.
 - 1.3. Após receber confirmação, os sensores deverão enviar sua leitura a cada 1s ao Gerenciador.
2. Atuadores:
 - 2.1. Todos os atuadores tem um identificador único.

- 2.2. Os atuadores devem se conectar ao Gerenciador e se identificar.
- 2.3. Os atuadores poderão ser ligados ou desligados pelo Gerenciador.
3. Gerenciador:
 - 3.1. O Gerenciador deverá aceitar a conexão de sensores e atuadores do sistema.
 - 3.2. O Gerenciador deve receber as leituras de todos os sensores do sistema e armazenar o último valor recebido.
 - 3.3. O Gerenciador deve ligar ou desligar os atuadores caso os valores das leituras de sensores indicarem que as variáveis estão fora dos valores máximo e mínimo configurados¹.
 - 3.4. O Gerenciador deve ser capaz de fornecer ao Cliente a última leitura de cada sensor do sistema quando receber uma requisição.
4. Monitoramento: o Cliente deve ser capaz de requisitar a última leitura de qualquer sensor do sistema ao Gerenciador.

¹Para diminuir a quantidade de atuações nos limites máximo e mínimo, é possível fazer um *controle por histerese*.

Aplicação 02: Incubadora Inteligente

Incubadoras são utilizadas em hospitais para proporcionar os bebês, geralmente prematuros, um ambiente termoneutro (isto é, o corpo não gasta energia compensando por mudanças na temperatura do ambiente) controlado pelo fluxo de ar, umidade e temperatura.

Nesta aplicação, as condições da incubadora são configuradas, monitoradas e controladas por um gerenciador, que se comunica com os sensores/atuadores e pode receber configurações ou responder consultas de um cliente externo.

Componentes

- Sensores:
 - Temperatura do ar
 - Umidade
 - Nível de Oxigenação
 - Batimentos cardíacos
- Atuadores:
 - Aquecedor (quando ligado, aumenta a temperatura do ar)
 - Umidificador (quando ligado, aumenta a umidade)
 - Circulador de ar (quando ligado, baixa a temperatura)
- Gerenciador, que funciona como o servidor da aplicação.
- Cliente, que pode requisitar valores das leituras dos sensores e recebe alarmes.

Requisitos Funcionais

1. Sensoriamento:
 - 1.1. Todos os sensores tem um identificador único.
 - 1.2. Os sensores devem se conectar ao Gerenciador e se identificar.
 - 1.3. Após receber confirmação, os sensores deverão enviar sua leitura a cada 1s ao Gerenciador.
2. Atuadores:
 - 2.1. Todos os atuadores tem um identificador único.
 - 2.2. Os atuadores devem se conectar ao Gerenciador e se identificar.
 - 2.3. Os atuadores poderão ser ligados ou desligados pelo Gerenciador.
3. Gerenciador:

- 3.1. O Gerenciador deverá aceitar a conexão de sensores e atuadores do sistema.
 - 3.2. O Gerenciador deve receber as leituras de todos os sensores do sistema e armazenar o último valor recebido.
 - 3.3. O Gerenciador deve ligar ou desligar os atuadores caso os valores das leituras de sensores indicarem que as variáveis estão fora dos valores máximo e mínimo configurados².
 - 3.4. O Gerenciador deve ser capaz de fornecer ao Cliente a última leitura de cada sensor do sistema quando receber uma requisição.
 - 3.5. O Gerenciador deve avisar ao Cliente (como um alarme) quando o nível de Oxigenação ou os batimentos cardíacos ficarem abaixo de um valor mínimo estabelecido.
4. Monitoramento:
- a) o Cliente deve ser capaz de requisitar a última leitura de qualquer sensor do sistema ao Gerenciador.
 - b) o Cliente deve ser capaz de receber alarmes do Gerenciador.

²Para diminuir a quantidade de atuações nos limites máximo e mínimo, é possível fazer um *controle por histerese*.

Aplicação 03: Sala de aula Inteligente

Nesta aplicação, temos uma sala de aula inteligente que permite a chamada de alunos e o desligamento automático dos aparelhos.

Componentes

- Sensores:
 - Presença de pessoas
 - Leitor de cartão dos alunos
 - Chave ON/OFF do projetor
- Atuadores:
 - Sistema de iluminação
 - Alimentador do projetor (controla o estado ON/OFF do projetor)
 - Alimentador do ar-condicionado (controle o estado ON/OFF do ar-condicionado)
- Gerenciador, que funciona como o servidor da aplicação.
- Cliente (professor)

Requisitos Funcionais

1. Sensoriamento:
 - 1.1. Todos os sensores tem um identificador único.
 - 1.2. Os sensores devem se conectar ao Gerenciador e se identificar.
 - 1.3. O leitor de cartão deve enviar ao Gerenciador o número e nome de cada aluno que confirmou presença em aula.
 - 1.4. O sensor de presença deve avisar o Gerenciador que há pessoas na sala.
 - 1.5. A chave do projetor deve notificar o Gerenciador toda vez que seu estado for alterado.
2. Atuadores:
 - 2.1. Todos os atuadores tem um identificador único.
 - 2.2. Os atuadores devem se conectar ao Gerenciador e se identificar.
 - 2.3. Os atuadores poderão ser ligados ou desligados pelo Gerenciador.
3. Gerenciador:
 - 3.1. O Gerenciador deverá aceitar a conexão de sensores e atuadores do sistema.

- 3.2. Quando forem detectadas pessoas, o gerenciador deve ligar o sistema de iluminação e o ar condicionado.
 - 3.3. Quando não houver detecção de pessoas por 15 minutos seguidos, o gerenciador deve desligar o sistema de iluminação, o projetor e o ar-condicionado.
 - 3.4. Quando a chave do projetor for ligada, o gerenciador deve apagar as luzes e ligar o projetor.
 - 3.5. Quando a chave do projetor for desligada, o gerenciador deve ligar as luzes e apagar o projetor.
 - 3.6. O Gerenciador deve guardar a lista de alunos que confirmaram presença na aula através do leitor de cartão.
 - 3.7. O Gerenciador deve desligar todos os equipamentos às 23h.
 - 3.8. O Gerenciador deve ser capaz de fornecer ao Cliente (professor) a lista de alunos que confirmou presença em aula.
4. O Cliente (professor) deve ser capaz de requisitar e receber a lista de alunos que confirmaram presença em aula (mínimo de 3 alunos para demonstração).