

**Universidade Federal de Pernambuco**  
IF975 - Redes de Computadores

**PROFESSOR:** Kelvin Lopes Dias ([kld@cin.ufpe.br](mailto:kld@cin.ufpe.br))  
**ESTÁGIO DOCÊNCIA:** Francisco Edigleison da Silva Barbosa ([fesb@cin.ufpe.br](mailto:fesb@cin.ufpe.br))  
**MONITOR:** Jean Silva ([jcns@cin.ufpe.br](mailto:jcns@cin.ufpe.br))

**Especificação do Projeto 2018.1**

- ✓ Número de integrantes em cada equipe: 3
- ✓ Data de lançamento do projeto: 18/04/18
- ✓ Data de entrega do projeto: 28 / 06 / 2018

## **1. Introdução:**

Este projeto é dividido em 3 questões, que estão relacionadas com temas abordados em sala de aula. As respostas devem ser anexadas no mesmo e-mail, juntamente com um relatório especificando o que foi feito em cada uma delas.

## **2. Regras:**

- a. Os códigos podem ser implementados em C, Java ou Python (altamente recomendado).
- b. O relatório final deve conter uma explicação bem detalhada do que está no código.
- c. Cópias acarretarão em 0 (Zero) para todas as equipes envolvida.
- d. O relatório deve ser entregue até 24h antes da apresentação.

## **3. Questões:**

### **I. Protocolos para comunicação (Camada de Transporte e Aplicação):**

- implementar um serviço que demonstre TCP, UDP e HTTP como mecanismos de transporte.
- Como sugestão, implemente uma aplicação cliente/servidor (Echo ou troca de mensagens).
- outra sugestão seria escrever o nome do protocolo num arquivo que seria lido pelo cliente e servidor para definir o modo de transmissão a ser utilizado.

## II. Armazenamento de arquivos (Cliente e Servidor):

- Deverá ser implementado um serviço de armazenamento de arquivo usando a arquitetura cliente/servidor
- O servidor tem que atender a mais de uma requisição (multithread)
- Deverá ser usado protocolo de transporte TCP
- Autenticação de Usuário
  - No primeiro acesso, o servidor cria uma pasta com o login do usuário;
  - Outras vezes que o usuário se loga, ele é direcionado para sua pasta;
  - Cada arquivo adicionado/removido atualiza um arquivo de texto / banco de dados (sqlite ou outros) com os arquivos que estão atualmente na pasta;
  - Comandos de listar arquivos fazer pesquisa nesse txt / banco de dados;
  - Comandos: GET copia/baixa arquivo; POST adiciona o arquivo; DELETE apaga o arquivo (isso pode ser feito com um método e dois parâmetros). Comando PUT é opcional e pode ser feito de várias formas, inclusive utilizando uma combinação de DELETE e POST.
- Funções do Serviço:
  - Upload e Download de arquivos;
  - Criar Pasta;
  - Compartilhar Arquivos e Pastas (um usuário tem acesso ao txt / BD do outro). Isso também pode ser resolvido com txt/BD.

## III. Jogo - Zerinho ou Um, Americano (P2P):

- Deverá ser implementado um jogo online modelo P2P usando comunicação via rede para disputas das partidas.
- Implemente um servidor ou proxy para gerenciar registro de jogadores e conexões entre peers.
- Implemente uma solução para problemas com perdas de conexão durante a partida.
- Ambos os jogadores devem receber uma mensagem de quem venceu a partida e a finalização da partida.
- Deverá ser usado protocolo de transporte UDP (experimente utilizar multicast)

- Sobre o Jogo:
  - Uma partida tem de 2 a 5 jogadores (Ids 0 a 4). Cada um ganha um ID
  - Todos mandam um valor entre 0 e 10 para o servidor, que soma os valores
  - Um algoritmo/método faz a contagem circular através dos Ids (Exemplo: <http://mapadobrincar.folha.com.br/brincadeiras/formulas-de-escolha/337-zero-um-americano>). O primeiro ID selecionado, vence.
  - Todos recebem os valores de todos, então cada um consegue determinar o vencedor.
  - Deve ser possível realizar partidas simultâneas (daí a importância do proxy ou multicast)

#### **4. Relatório Parcial**

- Apresentar um relatório parcial para cada uma das questões contendo as definições do protocolo a ser utilizado e módulos a serem desenvolvidos.
- O relatório da primeira questão deve conter detalhes do protocolo, incluindo: formato das mensagens e as ações a serem executadas.
- Não é para apenas repetir as funcionalidades esperadas como apresentadas acima, vocês devem apresentar as definições da equipe sobre como as questões serão implementadas.
- Evidentemente, ao longo da implementação podem acontecer mudanças nestas definições por questões não antecipadas inicialmente.

#### **5. Dicas:**

- Não deixe para começar o projeto mais tarde. Comece logo!
- É (quase) impossível fazer o projeto de “virada”... Mesmo em duas semanas de “viradas” ;-)
- Fazer um cronograma de atividades de desenvolvimento do projeto.
- Considere épocas de provas e o desenvolvimento de outros projetos em outras disciplinas;
- Não se esqueçam de dar atenção ao relatório! A entrega poderá ser feita 24h antes do dia da apresentação.
- O monitor acompanhará o desenvolvimento do projeto.
- A responsabilidade de acompanhamento do projeto é da equipe e não do monitor. Assim, não espere que o monitor “corra atrás” das equipes.

Bom Desempenho!