

Campus Ponta Grossa – DAINF

Bacharelado em Ciência da Computação

Sistemas Distribuídos

Prof. Dr. Richard Ribeiro

Sistemas Distribuídos

Projeto Final

Procedimentos (i.e., Contrato de Trabalho)

Projeto Final

#As entregas serão individuais.

#Os dados usados para testes devem ser padronizados logo no início do desenvolvimento;

#Deverão existir proteções contra dados inválidos:

>Recomendação: fazer um base de dados errados;

Projeto Final

- # Os itens das Entregas Parciais serão definidos pelo professor;
- # As datas dos TEPs e das EPs poderão ser adiadas, contudo:
 - > Isso só poderá ocorrer se for **antes** do dia previsto;
 - > Em comum acordo com **todos** os alunos presentes na aula;

Projeto Final

#Só receberá nota o aluno que for capaz de explicar qualquer pergunta feita pelo professor nas EPs;

#Se algum item da avaliação não funcionar quando conectar com outro programa (desenvolvido por outro aluno), o mesmo só receberá nota se funcionar com outro programa (de outra pessoa);

>Essa reavaliação do item só acontecerá após a avaliação de **todos** os outros alunos, e **apenas** se houver **tempo** disponível para tal.

Projeto Final

#Nas avaliações só poderá ser usado algum recurso externo ao campus (e ao laboratório usado nas mesmas) se o mesmo for acessível durante a avaliação por **todos** os outros alunos, em qualquer momento e sem uso de qualquer ajuda (como telefones celulares, p.ex.);

>Exemplos de tais recursos: nuvem interna ou externa ao campus (AWS, Google, Microsoft, Oracle, VPS (*Virtual Private Server*), etc.);

Etapas do Desenvolvimento

1. Discussão do tema e objetivos.
2. Definição de funcionamento.
3. Especificação de requisitos.
4. Definição do protocolo de troca de mensagens.
5. Construção do sistema cliente-servidor.

Atores

#Professor: gerente de projeto.

#Alunos: programadores.

Etapas

#Serão definidos prazos para cada etapa.

1. Discussão do tema e objetivos

#Prazo: 1 aula (máximo).

2. Definição de funcionamento

#Prazo: 1 aula (máximo).

3. Especificação de requisitos

#Prazo: 1 aula (máximo) para a definição.

#Deverão atender os requisitos mínimos estipulados pelo professor.

#Deverão estipular as regras que todos os clientes e servidores deverão seguir.

>Esta etapa deve ser feita entre os alunos e não precisa ser feita em sala de aula.

3.1. Requisitos Mínimos do PF (1)

- #Possuir pelo menos um módulo cliente e um módulo servidor.
- #Comunicação entre os módulos unicamente por troca de mensagens por *sockets* (TCP ou UDP).
- #Usar um protocolo de troca de mensagens definido em comum acordo com os alunos.

3.1. Requisitos Mínimos do PF (2)

#Um **cliente** deverá ser capaz de se comunicar com **todos** os **servidores**.

#Um **servidor** deverá ser capaz de se comunicar com **todos** os **clientes**.

4. Definição do protocolo de troca de mensagens

#Prazo: 2 aulas (máximo) para a escolha.

#Será iniciado em sala, mas a turma deverá terminar a definição do protocolo fora do horário de aula.

#O protocolo deverá estar disponível para todos os alunos e para o professor a todo momento.

#Será criado um PDF do protocolo um dia antes de uma Entrega Parcial (EPx) para ser usado na avaliação.

5. Construção do sistema cliente-servidor (1)

#Linguagem de programação Java.

#**Similaridade entre os códigos: aceitável até o limite de 50%.**

#Não poderão ocorrer correções no código no momento da Entrega Parcial.

5. Construção do sistema cliente-servidor (2)

#A partir da **EP-1** **todas** as mensagens enviadas e recebidas pelo Cliente e pelo Servidor **deverão** ser **mostradas** na saída padrão (**terminal**).

#A partir da **EP-2** o **Cliente** e o **Servidor** **deverão** possuir **interface gráfica**.

Entregas Parciais (1)

#Só receberá ponto o aluno **presente** na Entrega Parcial.

#O arquivo referente à uma EP deve ser enviado no máximo até o momento da avaliação. Sem ele o aluno não será avaliado.

#Se uma **entrega parcial** for **removida** os **pontos** referentes à **entrega (e ao teste) serão perdidos**.

Entregas Parciais (2)

#Os testes da entrega parcial **são obrigatórios**.

#Nos testes da entrega parcial **poderão** ser realizadas correções nos códigos fonte.

#Se **não existir o código fonte** referente a um item no **TEP**, o ponto relacionado ao referido item será zerado.

Entregas Parciais (3)

#Os itens avaliados em uma entrega parcial **deverão estar funcionando** nas entregas parciais subsequentes.

#Caso um item já avaliado anteriormente deixe de funcionar no momento de uma EP, a **pontuação** da entrega corrente será **descontada** do valor do item. Para o desconto será usado o valor do item da entrega anterior afetada.

#Mesmo que o item afetado **seja consertado posteriormente**, o valor subtraído **não será colocado de voltar na nota**.

Entregas Parciais (4)

#A entregas parciais **só ocorrerão** com o **envio prévio** para o Google Drive, e dentro do **prazo** estabelecido, do **protocolo de troca de mensagens**.

#Tal protocolo será impresso em PDF **um dia antes** da entrega parcial e será usado durante a mesma.

#**O aluno que não seguir o protocolo perde os pontos referentes ao item avaliado.**

Padronizações para o arquivo do PF

#Cada arquivo enviado deve ser nomeado: “NNNNNNNN-EPy.-zip”, onde “NNNNNNNN” representa o R.A. do aluno, e “y” o número da entrega parcial. Por exemplo:

>1234567-EP2.zip, mostra que o aluno com o R.A. 1234567 enviou o arquivo da Entrega Parcial 2.

Padronizações para o arquivo do PF

#Dentro do arquivo compactado enviado por cada aluno **deverá** existir um arquivo README.txt na raiz da estrutura com a explicação das ações necessárias para a **execução** correta do cliente e do servidor do aluno (biblioteca JSON, biblioteca DB, etc.).

ATENÇÃO!

#O não cumprimento de qualquer item definido nas padronizações impedirá a avaliação do aluno.

Dúvidas?