

Projeto POO

Carlos Olarte

18 de Outubro de 2019

Neste projeto vamos desenvolver um sistema que deve utilizar interfaces gráficas, comunicação entre os clientes e o servidor e armazenar as informações em arquivos ou bancos de dados (persistência).

1 REQUERIMENTOS

Antes de começar, escolha o sistema que gostaria de implementar. Por exemplo, você pode implementar uma (versão simplificada de uma) rede social (por exemplo, Facebook, Twitter, Instagram, etc.), um sistema de vendas por internet (Amazon, OLX, etc), um sistema de streaming de música (Spotify, etc), um sistema de registro acadêmico (SIGAA), etc.

Seguem os requerimentos básicos que seu sistema deve satisfazer.

1.1 SERVIDOR

O servidor deve permitir:

- Cadastrar novos usuários.
- Autenticar usuários: dados um login e uma senha, determinar se o usuário tem acesso ou não ao sistema.
- Armazenar os conteúdos / dados dos usuários, por exemplo, fotos de Instagram, mensagens de Twitter, vendas no Amazon, turmas no SIGAA, etc.

- Permitir realizar algumas operações do sistema implementado. Por exemplo, visualizar a lista de produtos na Amazon e realizar uma compra ; adicionar um post no Facebook; etc.

1.2 CLIENTE

O aplicativo utilizado pelos usuários deve permitir:

- Fazer login no sistema.
- Interagir com o sistema (depende do sistema implementado).

O aplicativo cliente deve utilizar uma interface gráfica de usuário (GUI). Também é permitido utilizar tecnologias web. Toda comunicação entre o cliente e o servidor deve utilizar sockets ou serviços web.

2 REGRAS

1. O trabalho é em duplas. Não é permitido a interação entre as diferentes duplas. Quaisquer infração a esta regra comprometerá a nota do projeto.
2. O código deve estar completamente documentado utilizando Pydoc.

2.1 PRAZOS E ENTREGAS

ter. 29/10, 12:00 (10%) Documento de requerimentos explicando claramente o sistema que vai ser construído. Lembre que você vai construir uma versão **simplificada** do sistema real. Portanto, neste documento, você deve deixar bem claro as funcionalidades do seu sistema. Você pode utilizar protótipos de interfaces gráficas para descrever melhor o que seu sistema vai fazer.

qui. 07/11 23:59 (20%) Diagrama de classes incluindo todas as classes que você vai implementar (e.g., interfaces gráficas, classes para manipular as informações dos usuários, etc). Você deve utilizar o padrão de design MVC.

seg. 18/11 23:59 (10%) Primeira entrega do código incluindo: 1) as classes básicas para manipular as informações dos usuários; 2) implementações básicas do servidor e do cliente.

qui. 28/11 12:00 (50%) Entrega final.

sex 29/11 10:50 (10%) Apresentação de um video de máximo 10 minutos que deve incluir:

- Descrição do sistema implementado;
- Ferramentas utilizadas;
- Funcionalidades implementadas ;
- Funcionalidades que não foram implementadas;

- O que aprendeu durante o processo ?
- Quais foram as maiores dificuldades ?
- Mostrar rapidamente como funciona o sistema.

3 CRONOGRAMA DE AULAS

Data	Assunto
Qua 23/10	Arquivos
Sex 25/10	Padrão de design MVC; JSON; Banco de dados
Ter 29/10	Entrega documento (12:00)
Qua 30/10	Sockets
Sex 01/11	Aula virtual: Biblioteca argparse
Qua 06/11	Aula virtual: Entrega diagrama de classes
Qui 07/11	Diagrama de classes (23:59)
Sex 08/11	Web services
Qua 13/11	GUI
Sex 15/11	Feriado Nacional
Seg 18/11	Primeira entrega (23:59)
Qua 20/11	Projeto (na minha sala)
Sex 22/11	Projeto (na minha sala)
Qua 27/11	Projeto (na minha sala)
Qui 28/11	Entrega final (12:00)
Sex 29/11	Apresentação videos