

**INSTITUTO FEDERAL DO ESPÍRITO SANTO – CAMPUS DE ALEGRE**

Curso: Tecnologia em Análise e Desenvolvimento de Sistemas	Disciplina: Programação 3	Turma: TADS 4
Professor: Cayo Magno da Cruz Fontana	Datas Entrega   Apresentações: 06/01//2025   Definidas no AVA	Valor: 40,00 pontos

**TS – TRABALHO SEMESTRAL**

**1. ESPECIFICAÇÃO**

A direção do IFES – Campus de Alegre – solicitou aos alunos da disciplina de Programação 3, do curso superior de Tecnologia em Análise e Desenvolvimento de Sistemas – TADS, o desenvolvimento de um sistema bibliotecário.

Para melhor compreensão do domínio do problema pelos programadores, o analista de sistemas construiu o diagrama de classes da solução proposta (disponível no Anexo I deste documento) e solicita a você, programador, que implemente este modelo de acordo as necessidades da instituição.

Deve-se construir um *software* utilizando a linguagem de programação Java, aplicando os conceitos e técnicas de Programação Orientada a Objetos.

Ao final, a solução construída deverá gerar uma solução de software semelhante a apresentada no seguinte vídeo (com áudio):

- <https://youtu.be/XwsbmrRSaY0>

**2. ATIVIDADES**

Analise o diagrama de classes presente no Anexo I deste documento e realize as necessidades a seguir:

1. Implemente todas as entidades (classes e interfaces) contidas no referido diagrama, organizando-as em seus respectivos pacotes;
2. As listas estáticas presentes nas classes `sisbli.acesso.Usuario` e `sisbli.biblioteca.Livro` devem armazenar, através de seus respectivos inicializadores estáticos, usuário e livros como forma de simular uma fonte de dados (SGBD) para este programa onde os dados são inteiramente voláteis.
3. O desenvolvedor deverá sentir-se à vontade na criação dos Usuários e dos livros. Contudo, deverá atentar-se acerca de:
  - a) Deverão haver ao menos uma instância de cada tipo de Usuário (classes hierarquicamente especialistas);
  - b) Deverão haver, ao menos 02 (dois) livros na biblioteca. A quantidade de exemplares, para cada um dos livros, é facultado ao programador.
4. O programa deve implementar o padrão de projeto (*Design Pattern*) Observador (*Observer*), através das interfaces `ReservouLivro` (produtor de eventos) e `LivroReservado` (consumidor de eventos), respectivamente. Sendo assim, o programador deve atentar-se na identificação da(s) classe(s) que produzirá(ão) o(s) evento(s) e da(s) classe(s) que consumirá(ão) o(s) eventos.
5. Ao ser executor, basicamente o programa deverá ser capaz de (ASSISTA O VÍDEO CUJO SEU LINK ESTÁ DESCRITO NO ITEM 1. DESTE DOCUMENTO):

- a) Realizar autenticação do usuário (qualquer usuário poderá autenticar-se no sistema. Contudo, para que as funcionalidades sejam, de fato, testadas para execução, naturalmente será o usuário do tipo `sisbli.biblioteca.Aluno` quem manipulará o sistema).
- b) Caso o usuário seja autenticado, será apresentado um menu para que ele consiga realizar as operações desejadas.
- c) Para os usuários que são instâncias de classes que implementam os produtores de eventos (padrão de projeto *Observer*), o menu deverá apresentar as seguintes opções:
  - Cadastrar Consumidor;
  - Remover Consumidor;
  - Cadastrar Reserva;
  - Sair.
- d) O usuário então deverá selecionar a opção desejada e produzir as ações necessárias para que a funcionalidade seja realizada.
- e) Ao final de cada ação realizada, o programa deverá retornar a tela do menu principal.
- f) Quando o usuário sair do sistema, todos os objetos que são instância de classe(s) consumidora(s) deverão exibir as suas respectivas informações de quando o evento é gerado pelo produtor.

NOTA1: o evento responsável por fazer com que o(s) produtor(es) notifiquem sua(s) respectiva(s) lista(s) de consumidor(es) é o cadastro da reserva de um livro! Contudo, os consumidores só notificarão os eventos ocorridos, para o sistema, quando o usuário finalizar o uso do sistema.

NOTA2: toda a interface gráfica com o usuário (GUI) deve ser realizada com o uso de "java popup's", disponíveis através da classe `JOptionPane` (pacote `javax.swing`).

### 3. SUBMISSÃO

Os trabalhos deverão ser submetidos, eletronicamente, para o email do professor, [cayo.fontana@ifes.edu.br](mailto:cayo.fontana@ifes.edu.br), com o título: "[TADS4] – Programação 3: Trabalho Semestral", contendo a seguinte estrutura:

Um arquivo compactado, agrupando todos os arquivos de código fonte (ou demais tipos, caso necessário) do projeto. O arquivo compactado poderá ter as extensões: 7z, zip, rar, gzip, bzip2 ou tar.

NOTA1: não incluir arquivos de ambientes de desenvolvimento – IDE's (como extensões .proj, .sln, .suo, entre outros).

PS 1: a cada dia de atraso, a nota do trabalho terá 5,0 (cinco) pontos reduzidos.

PS 2: trabalhos iguais receberão nota zero!

NOTA2: plágio é crime, previsto em legislação, e caracteriza uma conduta corrupta do cidadão! Contribua para uma sociedade menos corrupta! Diga NÃO ao plágio!!

### 4. APRESENTAÇÃO

O trabalho será apresentado em sala de aula (laboratório de informática), no formato pergunta-resposta entre o professor e o grupo, em uma das possíveis datas definidas neste documento (sorteio realizado em sala de aula). Contudo, o professor SEMPRE direcionará a pergunta para um componente (aluno) do grupo, por vez! Caso outro(s) aluno(s) manifeste(m)-se durante a entrevista de um componente específico, sem que esta interlocução seja previamente autorizada pelo entrevistador (docente), o grupo inteiro perderá ponto.

Os alunos utilizarão o computador do professor e apresentarão seus respectivos trabalhos no datashow. Os discentes não poderão utilizar quaisquer outros meios de consulta. As perguntas serão

exclusivamente sobre seus respectivos trabalhos que já estarão explicitamente projetados em sala, pelo datashow, no momento da sua apresentação.

NOTA: não poderá, em hipótese alguma, conter quaisquer comentários no código fonte deste trabalho. Caso haja, toda a nota de apresentação será invalidada pois, pela natureza da avaliação, a prática caracteriza fraude.

## **5. AVALIAÇÃO**

O trabalho será tecnicamente avaliado levando em consideração a realização todas as tarefas solicitadas no item 2. deste documento, enviado para o e-mail do professor até a data limite de submissão. Nesta etapa, o programa deverá estar compilando, executando e realizando todas as operações solicitadas no item 2. desta publicação.

No dia da apresentação o grupo deverá compilar, executar e testar as operações do programa. Para este último, o professor selecionará, aleatoriamente, quais operações serão desejadas para o teste.

Durante a entrevista, o entrevistador (professor) escolherá, aleatoriamente, algum aluno para responder uma determinada pergunta. Caso este aluno não saiba responder a pergunta proferida pelo professor, ou a resposta incorretamente, o grupo inteiro terá sua nota subtraída. O professor poderá oportunizar a um outro aluno (que também será escolhido aleatoriamente) para responder a pergunta que eventualmente não foi respondida pelo primeiro componente (ou que tenha sido respondida incorretamente), ou poderá passar para uma próxima pergunta para o mesmo ou outro componente do grupo.

## **6. PONTUAÇÃO**

A pontuação deste trabalho será dividida da seguinte forma:

- Avaliação de cada item definido no item 2. deste documento, enviado para o e-mail do professor até a data limite de submissão. (Valor: 16,00 pontos);
- No dia da apresentação, serão avaliados os seguintes quesitos (Valor: 24,00 pontos):
  - o Programa compilando com sucesso;
  - o Programa executando com sucesso;
  - o Programa realizando todas as operações que serão testadas aleatoriamente;
  - o Respostas corretamente claras, coesas e objetivas acerca do que será questionado na entrevista, pelo entrevistador (docente).

