



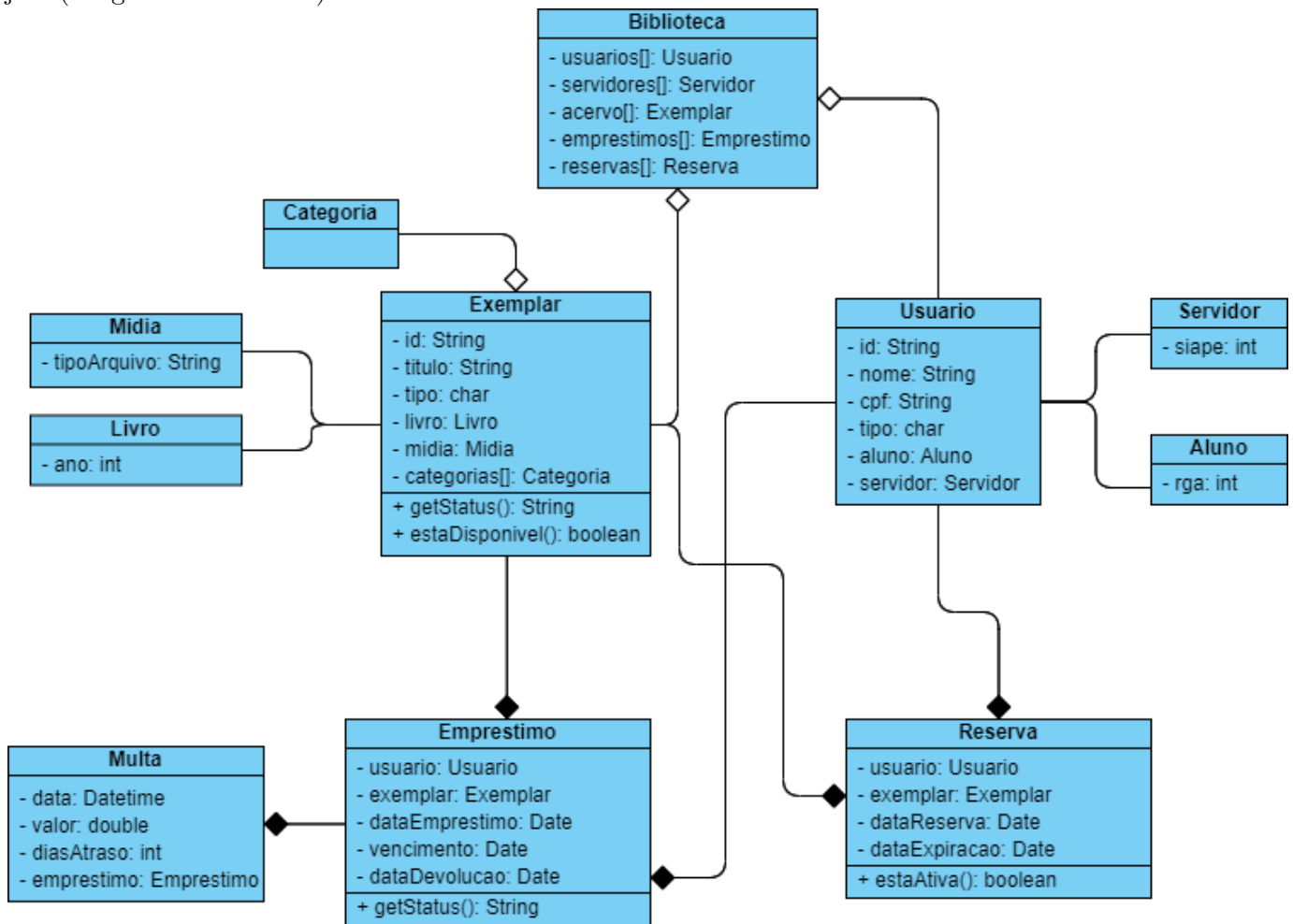
Aluno(a): \_\_\_\_\_ RGA: \_\_\_\_\_

## Trabalho 1 (14/05/2022)

### Descrição geral

Implemente um conjunto de classes para um sistema de controle de empréstimos de uma biblioteca.

A biblioteca “Círculo do saber” necessita de um sistema para manter e gerenciar informações sobre os exemplares do seu acervo, bem como realizar empréstimos e reservas dos exemplares. Para isso, elaborou-se o seguinte projeto (Diagrama de Classes):



Nesta biblioteca, há exemplares de diversos tipos, os quais podem ser emprestados aos usuários, desde que estejam na biblioteca e não tenham reservas. Os empréstimos realizados no sistema poderão ter três status diferentes: aguardando devolução, em atraso, devolvido no prazo, devolvido com atraso. Sempre que realizar um novo empréstimo, este deve ter o status “Aguardado devolução”.

Os exemplares devem ser identificados com o seu tipo, ou seja, se é um livro (L) ou uma mídia (M). A mesma identificação deve ser feita com os usuários, sendo (A) para aluno e (S) para servidor. Como há tipos diferentes nessas classes supracitadas, é necessário implementar dois construtores diferentes para cada uma das classes, de modo que cada construtor receba o objeto do seu tipo de dado correspondente.

A classe “Biblioteca” deverá conter os atributos para armazenamento das informações sobre os exemplares, os usuários, os empréstimos e as reservas. Recomenda-se que as funcionalidades estejam relacionadas a classes

específicas de acordo com o contexto.

## Interface de interação do sistema

Criar uma classe específica que contenha métodos para geração dos menus, bem como gerenciar a forma de entrada de dados no sistema pelo usuário. O gerenciamento da interface pode ser feito em mais de uma classe, caso seja necessário.

O acesso às funcionalidades do sistema se dará pelo menu principal:

- 1 - Gerenciar usuários
- 2 - Gerenciar exemplares
- 3 - Gerenciar categorias
- 4 - Gerenciar empréstimos
- 5 - Gerenciar reservas
- 6 - Consultar total de exemplares reservados por categoria em um período
- 7 - Sair do programa

Ao acessar a **opção 1** do menu principal, deve disponibilizar o seguinte menu:

- 1 - Cadastrar
- 2 - Consultar por CPF
- 3 - Excluir
- 4 - Editar
- 5 - Listar todos os cadastros
- 6 - Voltar

Ao acessar a **opção 2** e **opção 3** do menu principal, deve disponibilizar o seguinte menu:

- 1 - Cadastrar
- 2 - Consultar por código
- 3 - Excluir
- 4 - Editar
- 5 - Listar todos os cadastros
- 6 - Voltar

Ao acessar a **opção 4** do menu principal, deve disponibilizar o seguinte menu:

- 1 - Empréstimo
- 2 - Consultar status de um empréstimo
- 3 - Devolver
- 4 - Renovar
- 5 - Listar todas os empréstimos
- 6 - Voltar

Ao acessar a **opção 5** do menu principal, deve disponibilizar o seguinte menu:

- 1 - Reservar
- 2 - Consultar status de uma reserva
- 3 - Cancelar
- 4 - Listar reservas ativas para um exemplar
- 5 - Voltar

O retorno dos métodos relacionados ao cadastros devem corresponder aos objetos referentes a cada funcionalidade. Por exemplo, o método para cadastrar usuários, deve-se possibilitar que se entre com as informações dos usuários da biblioteca e, ao final, deve ser retornar um novo objeto do tipo usuário.

Observe a separação entre as classes de negócio e a classe de interface. As classes de negócio não devem possuir nada que crie uma dependência da interface. Desta forma, se posteriormente você decidir mudar a interface, por exemplo, usar uma interface gráfica, as classes de negócio não precisam ser alteradas. A classe Biblioteca irá atuar como uma classe gerente, fazendo a interligação das classes de negócio com a classe de interface. Em um projeto mais profissional, provavelmente existiriam várias interfaces e classes gerente.

## Requisitos

- Deve-se priorizar a declaração de todos os atributos com acesso privativo. Atributos que forem declarados com outros modificadores de acesso deverão ter breves comentários justificando a escolha.
- O usuário e/ou exemplar não pode ser excluído se existir algum empréstimo ou reserva para ele.
- O empréstimo de um exemplar somente poderá ser realizado (i) se o item não estiver com um empréstimo ativo, (ii) se não existir reservas e (iii) se houver reservas, estas devem estar no nome do usuário que está requerendo o empréstimo.
- Pode haver mais de uma reserva para o mesmo exemplar.
- A ordem de reservas deve ser respeitada no momento do empréstimo.
- A data de expiração da reserva deve ser calculada de acordo com a situação do exemplar: (i) se estiver disponível, a data deve ser a atual acrescida de um dia; (ii) se estiver emprestado, a data deve ser o vencimento acrescido de um dia; (iii) se já houver reservas, a data deve ser a expiração da última reserva acrescida de um dia.
- O usuário poderá ter apenas uma reserva ativa em um determinado período (intervalo entre a data da reserva e a data de expiração).
- A renovação apenas pode acontecer se não houver reservas previstas para o exemplar e o empréstimo não estiver em atraso.
- A quantidade de dias para reserva e renovação devem ser campos estáticos e dependem do tipo de usuário.
- O valor por dia de atraso deve ser um campo estático, sendo o valor da multa calculado sobre este valor multiplicado pela quantidade de dias que o objeto foi entregue em atraso.
- Usuários com empréstimos atrasados não podem realizar reservas e nem novos empréstimos.
- As entradas e saídas ficam a cargo do grupo (terminal ou interface gráfica – janelas de pop-up).
- O atendimento dos campos listados na especificação de cada funcionalidade deve ser seguido.
- Comentários sobre o funcionamento dos métodos, com exceção de *getters*, *setters* e construtores deverão estar presentes no código-fonte.
- Não serão contabilizados na nota comentários óbvios como:

```
//Este método calcula a média  
public double calculaMedia()
```

Considere descrever de forma mais específica:

```
/* Este método percorre cada disciplina cadastrada para um aluno,  
e para cada disciplina, percorre todas as notas cadastradas,  
acumula as notas e então calcula a média */  
public double calculaMedia()
```

## Critérios de avaliação

O trabalho deverá ser entregue no AVA até o dia 14/05/2021. Deve ser entregue um arquivo compactado contendo o projeto do Netbeans (ou outra IDE que esteja utilizando). A avaliação será dividida em 8 (oito) etapas e em cada etapa será avaliada a codificação e a execução. As etapas são:

<b>Critério</b>	<b>Pontuação</b>
Biblioteca e classes de funcionalidades	1,5
Usuário	1,0
Acervo	1,5
Empréstimos	1,5
Reservas	1,5
Interface	1,0
Relatórios de empréstimos em um período	1,5
Comentários	0,5