

Considere um sistema de biblioteca simplificado definido pela seguinte interface:

```
public interface SistemaBiblioteca {
    public boolean pegarLivro(String matriculaUsuario, String idLivro)
        throws UsuarioInexistenteException, LivroInexistenteException;
    public boolean devolverLivro(String matriculaUsuario, String idLivro)
        throws UsuarioInexistenteException, LivroInexistenteException;
    public boolean cadastrarUsuario(String matricula, String nome, TipoUsuario tipo)
        throws UsuarioJaExisteException;
    public void cadastrarLivro(String id, String titulo, List<String> autores)
        throws LivroJaExisteException;
    public Map<String, Livro> getTodosOsLivros();
    public Collection<Usuario> getUsuarios();
    public Collection<Livro> getLivrosEmprestadosA(String matriculaUsuario);
}
```

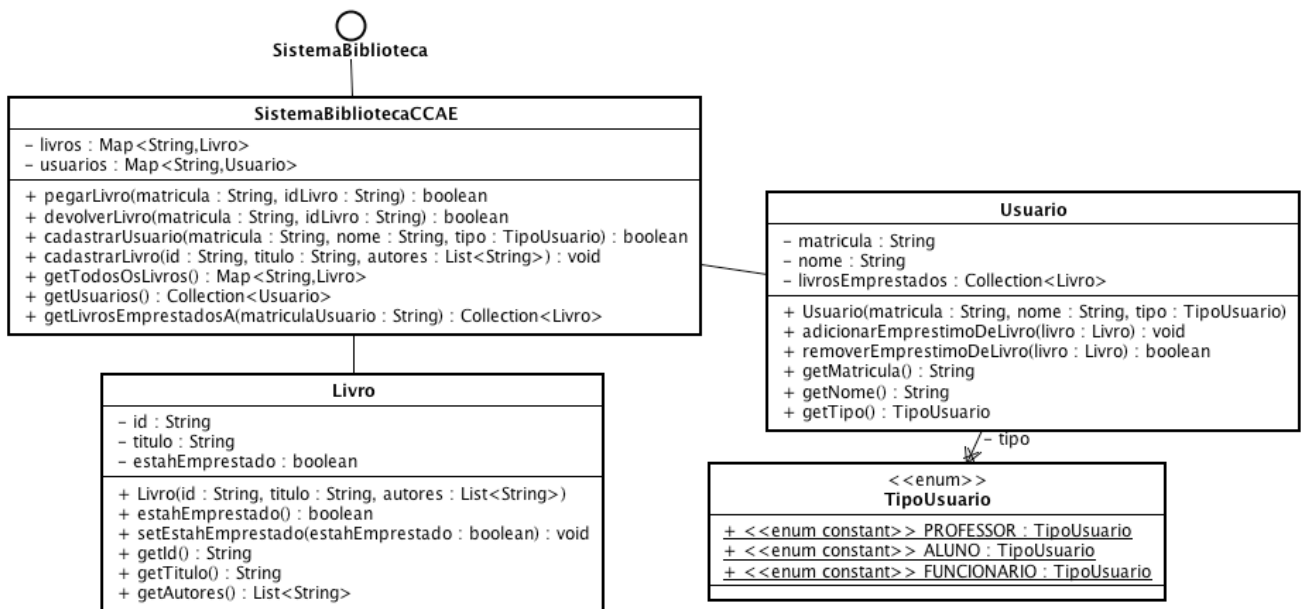
Como se pode ver, o empréstimo pode gerar exceções. Se na hora de pegar um livro, ele já estiver emprestado, o retorno do método é `false`. Uma das classes que implementa essa interface é a classe `SistemaBibliotecaCAAE`, cujo esqueleto é mostrado abaixo:

```
public class SistemaBibliotecaCAAE implements SistemaBiblioteca {

    private Map<String, Livro> livros = new HashMap<String, Livro>();
    private Map<String, Usuario> usuarios = new HashMap<String, Usuario>();
    ...}

```

Considerando isto e o digrama abaixo, resolva as seguintes questões:



1. Implemente o método `pegarLivro` da classe `SistemaBibliotecaCAAE`. Veja que essa ação implica em alterar o status de `estahEmprestado` do `Livro` e em adicioná-lo à coleção dos `livrosEmprestados` do `Usuario` cuja matrícula é passada. Observe bem os métodos dessas classes.
2. Implemente um teste para o método `cadastrarLivro` da classe `SistemaBibliotecaCAAE`. Este teste deve verificar se antes de cadastrar, o livro não se encontra no sistema e depois deve observar se o livro foi cadastrado corretamente, conferindo seu `id` e seu `titulo` nos livros já cadastrados no sistema.