

## Controle e acompanhamento de exames médicos

O objetivo deste desafio é desenvolver um protótipo de um sistema para controlar e acompanhar a realização de exames médicos. A solução deverá contemplar os requisitos e situações descritas abaixo. Os pacientes chegam ao laboratório com uma requisição na qual são solicitados diversos exames, como colesterol, glicose e diversos outros. Na requisição é identificado o médico que fez a solicitação. Existem diferentes funcionários no laboratório, o responsável por lançar as requisições e exames, responsável pela coleta e quem elabora o parecer e diagnóstico do exame.

Após o lançamento da requisição no sistema, com todos os detalhes, descrição do exame, data e outros dados, a coleta é realizada e o responsável faz o registro do material coletado, a quantidade (ml) e a hora em que foi coletado. O material pode ser sangue, urina, etc.

Quando o exame é finalizado o responsável pelo diagnóstico faz o registro dos resultados, o parecer e outras informações pertinentes.

É fundamental que o sistema tenha um controle de usuários, portanto, quando o usuário acessa o sistema e escolhe a ação que vai realizar, é necessário que sejam validados os privilégios dele. Podem existir usuários que fazem mais de um papel, por exemplo, que lança requisições e exames também pode, eventualmente registrar coleta e assim por diante. O sistema deve ter um ou mais usuários administradores que podem fazer tudo, inclusive criar usuários e definir papéis.

Os exames serão armazenados pelo sistema, permitindo a emissão de relatórios e consultas. O paciente pode acessar o sistema e consultar os resultados de um exame, mas somente os seus exames. Deve existir um relatório que mostre todos os exames feitos em um período. Deve ser mostrada a requisição e para cada requisição, os dados de todos os exames.

Outros relatórios deve mostrar estatísticas de exames, quantos exames foram feitos para cada tipo diferente (colesterol, plaquetas, etc), contagem por paciente e por médico requisitante.

Implementação (critérios e características):

- Utilizar fundamentos da orientação a objetos, como encapsulamento, herança, polimorfismo, etc.
- Pelo menos uma classe deve implementar herança múltipla, herdando de uma classe e implementando uma interface.
- Armazenar os dados em memória ou arquivo (o grupo pode escolher).

Entregar o diagrama de classes completo, com todas as classes, atributos e métodos.