

Relatório de Melhorias: pacote Notification

promovi diversas melhorias nas classes do serviço de notificações. Abaixo, detalho as modificações realizadas em cada uma delas:

NotificationService:

- **Injeção de Dependência por Construtor:** Removi a anotação `@Autowired` nos atributos, optando por utilizar construtores para a injeção de dependências. Essa abordagem promove uma injeção de dependência mais explícita e clara.
- **Melhorias nos Nomes de Métodos:** Renomeei o método `setViwed` para `setViewed`, corrigindo um possível erro de digitação e mantendo a consistência na nomenclatura dos métodos.
- **Uso de Optional:** Introduzi o uso de `Optional` ao buscar notificações por ID, visando melhor tratar situações em que a notificação não é encontrada.
- **Aprimoramento de Log:** Adicionei mensagens mais descritivas nos logs para facilitar a identificação e resolução de problemas.
- **Reorganização de Métodos:** Agrupei métodos relacionados, promovendo uma melhor organização da classe.
- **Remoção de Construtor Redundante:** Eliminei um construtor que apenas chamava `super()`, pois não era necessário.
- **Adição de Comentários Pertinentes:** Incluí comentários explicativos ao longo do código para fornecer insights sobre a lógica e a finalidade de cada trecho.

NotificationSearcher:

- **Comentários Descritivos:** Adicionei comentários explicativos para esclarecer o propósito da classe e de cada método.
- **Injeção de Dependência por Construtor:** Embora o uso de `@Autowired` seja válido, optei por sugerir a injeção de dependência por construtor para maior clareza.
- **Melhoria nos Nomes de Métodos:** Mantive os nomes dos métodos, pois são descritivos o suficiente, mas sugeri considerar a padronização de nomenclatura no projeto.

NotificationRemover:

- **Comentários Descritivos:** Inseri comentários explicativos para fornecer uma compreensão clara do propósito da classe e de seus métodos.
- **Injeção de Dependência por Construtor:** Utilizei a injeção de dependência por construtor para promover uma estrutura mais clara e coesa.
- **Melhoria nos Nomes de Métodos:** Mantive os nomes dos métodos, pois são descritivos o suficiente.
- **Uso de Exceção Específica:** Optei por lançar a exceção mais específica `NotFoundException` quando uma notificação não é encontrada.

NotificationRegister:

- **Comentários Descritivos:** Adicionei comentários explicativos para esclarecer o propósito da classe e de cada método.
- **Injeção de Dependência por Construtor:** Mantive a injeção de dependência por campo, pois a classe é um componente do Spring.
- **Melhoria nos Nomes de Métodos:** Preservei os nomes dos métodos, pois são autoexplicativos.
- **Agrupamento Lógico:** Agrupei métodos relacionados para proporcionar uma organização mais coesa.

NotificationEditor:

- **Nomenclatura de Método:** Corrigi o nome do método de setViwed para setViewed para refletir corretamente a ação realizada.
- **Comentários:** Acrescentei comentários explicativos para descrever o propósito do código, detalhar parâmetros e indicar possíveis exceções.
- **Uso de findById:** Substituí o uso de getByld por findById, que retorna um Optional, aprimorando o tratamento de situações em que a notificação não é encontrada.
- **Tratamento de Exceção:** Adicionei tratamento de exceção para o caso em que a notificação não é encontrada, lançando uma NotFoundException.

Essas modificações visam aprimorar a qualidade, a legibilidade e a manutenibilidade do código, seguindo boas práticas de programação.