**Design Patterns – Observer**

Observer é um padrão de projeto, classificado como um padrão comportamental, que permite definir uma classe onde possibilita que, quando um objeto muda de estado, todos os outros objetos interessados são notificados e atualizados automaticamente.

**Observer no MusicAll**

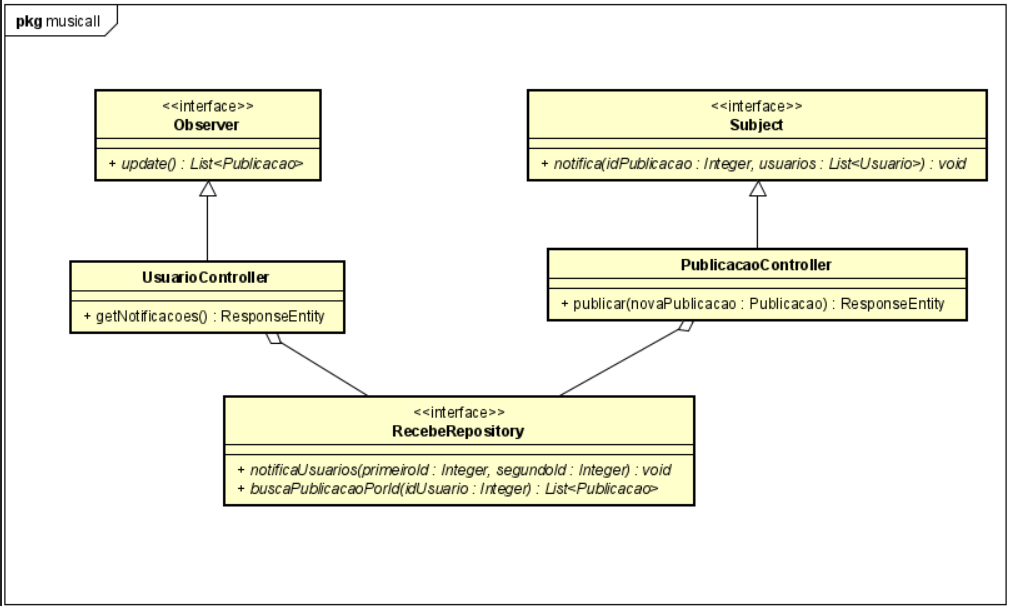
Interface Observer 🡪 Interface observadora

Interface Subject 🡪 Interface observada

Classe Publicação 🡪 Classe observada

Classe Usuário 🡪 Classe observadora

A classe **Usuário** irá observar a classe Publicação, verificando se há alguma alteração através da interface **RecebeRepository**. Caso tenha alguma mudança, o Observer irá consultar a lista de usuários cadastrados e notificar a mudança.



**Exemplo 1:** A usuária Luana está muito animada e gostaria de encontrar logo uma banda! Por conta disso, gostaria de ser notificada sempre, sobre novas publicações de novas bandas. Assim, a classe Observer, observa a classe Publicações e se perceber mudanças, ou seja, novas publicações, a Luana será notificada.

**Exemplo 2:** O usuário Silva faz uma publicação dizendo que vai formar uma banda com o gênero musical MPB. Essa alteração ocorre na classe Publicação e é notada pelo o Observer, que analisará a lista e notificará os usuários que possuem interesse em participar das bandas com o mesmo gênero musical.

**Porque utilizar o Observer?**

Utilizando o padrão de projeto Observer, é possível filtrar as pessoas que querem ser notificadas sobre novas publicações e as que não querem, evitando as notificações indesejadas e tornando as publicações mais relevantes para o publicador.