



11

Opcional: Validando o usuário no serviço Rest

Você é um desenvolvedor back-end de uma startup que possui uma aplicação *web* de anúncios de restaurantes e lanchonetes. A aplicação que foi desenvolvida usando *AngularJS* no client-side consumindo serviços *Rest* desenvolvidos em *Jersey/JAX-RS*, precisa fazer um cadastro dos seus usuários que possuem um `username`.

Esse `username` possui regras de quais caracteres ele pode compor e nós queremos que ele seja válido antes de consultarmos no banco de dados as informações sobre ele.

Vamos ver abaixo a *action* que receberá o usuário para ser cadastrado:

```
@POST
@Path("/user")
@Consumes(MediaType.APPLICATION_JSON)
public Response listaDeRestaurantes(@Valid User user) {
    // codigo omitido
}
```

O projeto utiliza a especificação *Bean Validation* que permite realizar validações de modelos através de anotações. Para validar um campo usando expressões regulares, podemos usar a anotação `@Pattern(regex="...")`

```
// User.java
public class User {
    @Pattern(regex = "???")
    @NotEmpty
    private String username;
}
```

O `username` precisa ser da seguinte forma:

- O limite é de 10 caracteres;
- O primeiro caractere deve ser uma letra do alfabeto, não pode ser um número;
- A partir do segundo caractere podemos ter letras maiúsculas, minúsculas e números;

Como deve ficar a anotação `@Pattern` com uma expressão regular com essas características?

Responda

| INSERIR CÓDIGO | FORMATAÇÃO |
|--|------------|
| <pre>@Pattern(regex="EXPRESSAO-REGULAR-AQUI") @Path("/user") @Consumes(MediaType.APPLICATION_JSON) public Response listaDeRestaurantes(@Valid User user) { // codigo omitido }</pre> | |

