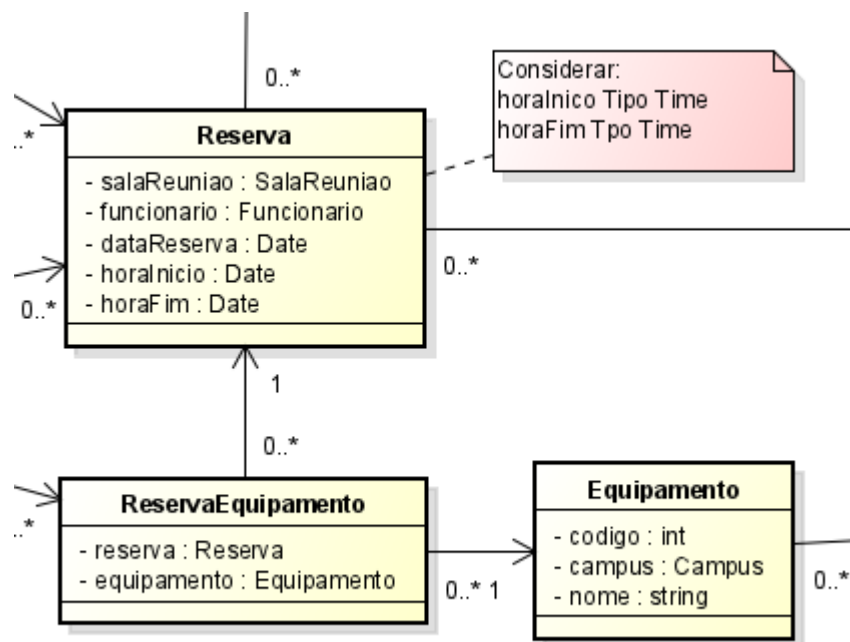


Como o projeto foi melhorado e onde foram aplicado os princípios GRASP

Foi usado no desenvolvimento do trabalho os seguintes princípios GRASP :

- Indireção
- Baixo Acoplamento
- Especialista
- Criador

Para fazer a reserva de equipamentos no lugar de ter uma classe chamada de “ReservaEquipamento” foi preferido reduzir o acoplamento e a responsabilidade dela, pois até então ela deveria ser responsável pelo equipamento , pelos horários de reserva deles e as listas de reservas.



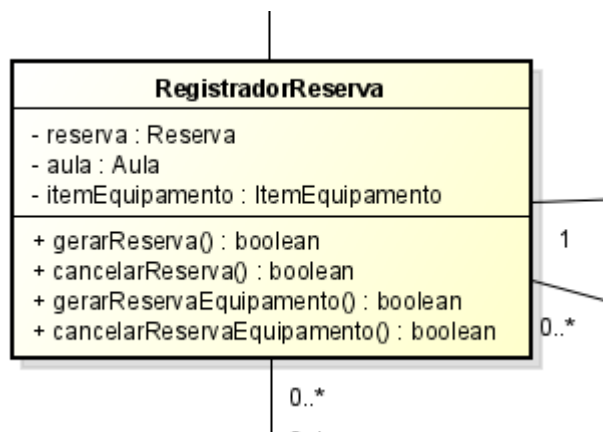
Nesse modelo a classe “Reserva” é responsável apenas pela criação de uma reserva incluindo horários , data início e fim e um local sala e um funcionário , ela não sabe da existência da classe “Equipamento”.

A classe “Equipamento” é responsável apenas por definir o equipamento no sistema e ela não tem nenhum conhecimento da classe Reserva, no caso ela não tem nenhuma chave estrangeira além de “Campus” ao qual ele pertence

Foi criado uma classe “ItemEquipamento” que serve para fazer a relação do Equipamento com a sua Reserva, essa classe contém ambas as chaves estrangeiras e isso permite os seguintes princípios GRASP : Indireção pois ela relaciona a classe equipamento e reserva e isso faz com que não exista uma ligação direta entre essas classes . Baixo Acoplamento

pois as classes tendo atributos privados próprios sem a necessidade de alterar comportamento direto em outras classes faz com que eu possa usar por exemplo : classe “Equipamento” em outra parte do código sem que ela tenha qualquer relação direta com “Reserva” . E Especialista , no caso a classe “ItemEquipamento” é especialista em criar uma reserva de um equipamento , essa classe é quem será chamada pela classe gerenciadora.

A Classe “RegistradorReserva”



Ela é uma classe Criador pois ela é quem instancia um novo objeto “Reserva” para a classe “ItemEquipamento” criar o relacionamento de Reserva e Equipamento, no caso os equipamentos já estão inseridos no banco antes de ser criado uma nova reserva deles.