



**GOVERNO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA
PARAÍBA
CAMPUS CAMPINA GRANDE
CURSO BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO
DISCIPLINA: PADRÕES DE PROJETOS
PROFESSOR: KATYUSCO DE FARIAS SANTOS**

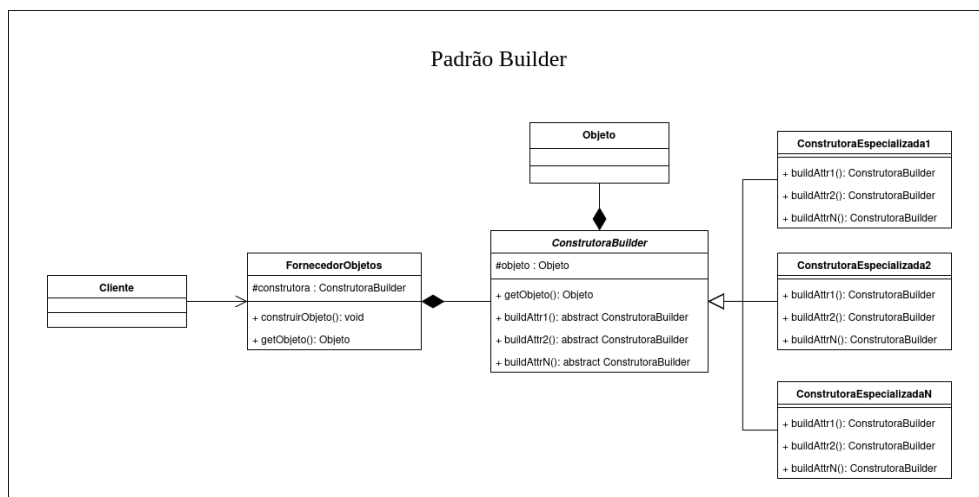
**Diferenças entre os padrões de projeto para a construção de objetos *Builder*,
Factory Method e *Abstract Factory***

**Aluno:
Marcus Cauê de Farias Barbosa - 202321250026**

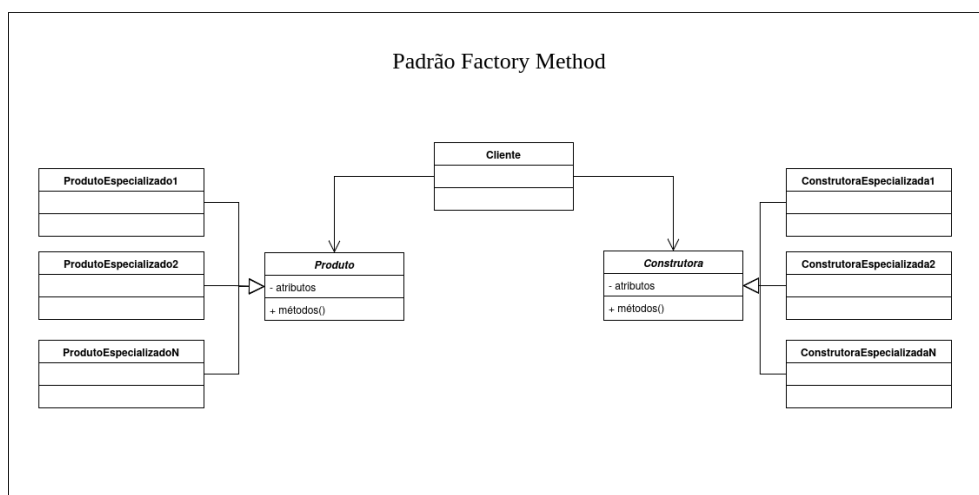
**Campina Grande
Junho / 2025**

Todos os padrões de construção servem para que o cliente não precise se preocupar com detalhes de implementação dos objetos, mas que apenas possa fazer uso deles. Também contribuem para que os objetos “produto” possam se desenvolver (crescer em hierarquia, atributos e funcionalidades) independentemente das classes “construtoras”.

Quando um objeto é construído através do padrão *Builder*, o cliente o obtém através de um “fornecedor de objetos”, o qual está relacionado com um “construtor de objetos” genérico que pode se especializar a depender do tipo de produto especializado que o cliente requisita. Para fazer a construção do objeto, o “construtor” define métodos para a instanciação de cada um dos atributos do objeto, de modo que as regras de negócio para os atributos possam estar modularizadas e possam ser definidas pelos “construtores especializados”. Ou seja, a construção do objeto ocorre em etapas e um objeto especializado é obtido pelo cliente quando ele informa para o fornecedor de objetos qual tipo de “construtor especializado” “fabrica” o objeto que ele quer.



De sua parte, o padrão *Factory Method* possui algumas similaridades ao padrão de interface *Bridge*, em que ele desacopla a hierarquia de classes do objeto produto da hierarquia de classes construtoras, de modo que ambas possam se desenvolver independentemente. Para o cliente obter o objeto, ele precisa solicitá-lo ao construtor específico que o “fabrica”. Nesse padrão, sempre que um novo produto for adicionado à hierarquia, será necessário também adicionar uma classe construtora para ele.



Por fim, o padrão *Abstract Factory* busca solucionar um problema do *Factory Method*, que é o crescimento linear da quantidade de classes produto e classes construtoras. Em outras palavras, toda vez que se adiciona um novo produto, uma classe construtora também é adicionada e isso pode ser problemático se o domínio do problema tiver muitos produtos. A abordagem que o *Abstract Factory* segue para resolver esse desafio é tratar os produtos e as classes construtoras em conjuntos, em grupos. Sendo assim, quando o cliente solicita um objeto, ao invés de pedir diretamente qual produto específico ele quer, ele pede que a construtora específica forneça o objeto específico que ela constrói, o qual, de sua parte, pertence a um grupo específico. Assim fica mais fácil de tratar os produtos específicos, porém é necessário ter controle sobre os grupos de produtos, e sempre que um novo grupo é criado, cada “fabricante” precisa implementar um método para a construção de um objeto específico que pertença a esse grupo.

