

Arquitetura

A solução implementa uma arquitetura de binding de dados utilizando o padrão Observer, a arquitetura é composta pelas seguintes classes:

- **Observable:**
 - Gerencia uma lista de observadores e notifica-os sobre mudanças.
- **Observer:**
 - Interface que define um método update para ser implementado por qualquer classe que queira observar mudanças.
- **BindingManager:**
 - Gerencia o vínculo entre o modelo e a visão, configurando bindings unidirecionais e bidirecionais.
- **DesafioDataBinding:**
 - Faz o teste do modelo.

Padrão Observer

O padrão Observer é um padrão de design comportamental que define uma dependência um-para-muitos entre objetos, onde quando um objeto muda de estado, todos os seus dependentes são notificados e atualizados automaticamente. Nesta implementação:

- **Observable** é a classe que guarda o estado e notifica os observadores sobre qualquer mudança.
- **Observer** é a interface que define o contrato para os observadores que desejam ser notificados sobre mudanças no estado do Observable.

Configurações de Binding Suportadas

- **One-Way Binding (Binding Unidirecional):**
 - Permite a sincronização de dados de um modelo para uma visão ou vice-versa. Dependendo da direção configurada (modelToView), a atualização pode ser do modelo para a visão ou da visão para o modelo.
- **Two-Way Binding (Binding Bidirecional):**
 - Permite a sincronização de dados tanto do modelo para a visão quanto da visão para o modelo. Qualquer mudança no modelo ou na visão será refletida na outra parte.