

PÓS GRADUAÇÃO DESENVOLVIMENTO WEB 2020-2021

O USO DE PADRÕES DE PROJETO
NO GERENCIAMENTO DE UMA MATRIZ DE RISCO

OBJETIVO

Demonstrar como a Orientação a Objetos seguida do uso de Padrões de Projeto podem contribuir estruturalmente para uma problemática onde se tem várias situações transitórias dependentes de um dado inserido externamente.

PROBLEMA?

O PROCESSAMENTO DE UMA MATRIZ DE RISCO

**MATRIZ DE RISCO PARA MONITORAMENTO
ESTRATÉGICO DO DISTANCIAMENTO**

TAXA DE POSITIVIDADE¹	MUITO CRÍTICA > 70%	RISCO MODERADO	RISCO ALTO	RISCO ALTO	RISCO MUITO ALTO	RISCO EXTREMO	RISCO EXTREMO
	CRÍTICA 61% A 70%	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO EXTREMO	RISCO EXTREMO
	MUITO ALTA 51% A 60%	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	ALTA 41% A 50%	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	MODERADA 31% A 40%	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	BAIXA 21% A 30%	RISCO BAIXO	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	MÍNIMA ≤ 20%	RISCO BAIXO	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
		MÍNIMA ≤ 50%	BAIXA 51% A 60%	MODERADA 61% A 70%	ALTA 71% A 80%	MUITO ALTA 81% A 90%	CRÍTICA > 90%
TAXA DE OCUPAÇÃO GERAL DE UTI ADULTO (PÚBLICO E PRIVADO)							

¹ Percentual de Exames Positivos / Exames Realizados (Síndrome Gripal + Síndrome Respiratória Aguda Grave)

ESTADOS TRANSITÓRIOS

- Taxa de Positividade
- Taxa de UTI
- Risco

MATRIZ DE RISCO PARA MONITORAMENTO ESTRATÉGICO DO DISTANCIAMENTO							
TAXA DE POSITIVIDADE¹	MUITO CRÍTICA > 70%	RISCO MODERADO	RISCO ALTO	RISCO ALTO	RISCO MUITO ALTO	RISCO EXTREMO	RISCO EXTREMO
	CRÍTICA 61% A 70%	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO EXTREMO	RISCO EXTREMO
	MUITO ALTA 51% A 60%	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	ALTA 41% A 50%	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	MODERADA 31% A 40%	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	BAIXA 21% A 30%	RISCO BAIXO	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	MÍNIMA ≤ 20%	RISCO BAIXO	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
		MÍNIMA ≤ 50%	BAIXA 51% A 60%	MODERADA 61% A 70%	ALTA 71% A 80%	MUITO ALTA 81% A 90%	CRÍTICA > 90%
TAXA DE OCUPAÇÃO GERAL DE UTI ADULTO (PÚBLICO E PRIVADO)							

¹ Percentual de Exames Positivos / Exames Realizados (Síndrome Gripal + Síndrome Respiratória Aguda Grave)

SOLUÇÕES EM PROGRAMAÇÃO

- **Programação estruturada**
- **Programação Orientada a Objetos**

PROGRAMAÇÃO ESTRUTURADA

SEQUÊNCIAS

```
<?php
```

```
$numero1 = 18  
$numero2 = 20  
$total = $numero1 + $numero2  
echo $total
```

DECISÃO

```
<?php
```

```
$idade = 22;  
if ($idade >= 18) {  
    echo "Maior de idade"  
}
```

FUNÇÕES

```
<?php
```

```
function soma($var1, $var2) {  
    $resultado = $var1 + $var2  
    return $resultado  
}  
echo soma(20,40)
```

PROGRAMAÇÃO ORIENTADA A OBJETOS

- Abstração: Defini os contornos de um objeto;
- Indução de tipos: Tipagem de dados usada como uma abstração;
- Encapsulamento: Proteger as implementações de uma classe;
- Visibilidade: Private, Protected e Public;
- Getters e Setters: Prove acessibilidade preservando o encapsulamento
- Herança: Uma classe estende as funcionalidades de outra
- Polimorfismo: Objetos com a mesma interface podem realizar diferentes tarefas

PROGRAMAÇÃO ORIENTADA A OBJETOS

Divisão de problemas, modularização e classes

Dividir um problema complexo pode transformá-lo em vários subproblemas simples, mas é necessária uma maneira de organizar os módulos e trabalhar com eles uns em relação aos outros, para lidar com o problema maior sendo resolvido. Esses módulos são as **classes**.

O processo de decomposição de um problema em subproblemas menores corresponde ao processo de **modularização**.

PROGRAMAÇÃO ORIENTADA A OBJETOS

Como os paradigmas trabalham

Assim como na programação orientada a objetos, a programação estruturada usa **modularidade e reuso**. No entanto, a programação procedural não oferece classes, nas quais as tarefas de programação podem ser agrupadas em objetos.

Objetos de classes podem operar com suas próprias estruturas de dados e isso não pode ser feito apenas pelas funções. A programação procedural exige sequências longas para executar tarefas maiores.

PADRÕES DE PROJETO

Porque usar orientação a objetos e padrões de projeto

- Resolver problemas menores e reunir tudo em uma única solução mais ampla;
- Alterar um módulo sem provocar erros em todo o programa;
- Agilizar o trabalho em equipe;
- Respeito ao principio da responsabilidade única;
- Desacoplamento entre design e implementação;
- Favorecimento da composição e da delegação;
- Velocidade de revisão e atualização;

PADRÕES DE PROJETO

Características - Um solicitante para as informações

Veremos uma classe “Cliente”, e que esta tem diferentes papéis em projetos maiores, mas o papel principal é fazer **solicitações a partir de classes que constituem o padrão de projeto.**

Ao permitir que o cliente use os dados da forma mais genérica possível, a classe poderá fazer o que quiser com esses dados.

PADRÕES DE PROJETO

Características - Programando para abstrações

Em padrões de projeto programa-se para uma interface e não para sua implementação. Isto significa que a **definição de variáveis correspondem à uma instância de um tipo de dados correspondente a uma classe abstrata ou uma interface** em vez de utilizar-se uma implementação concreta.

PADRÕES DE PROJETO

Características - Sobre velocidade

Nas situações em que a velocidade de revisão é importante, seu programa deve levar em conta a velocidade tanto da operação quanto do desenvolvimento. **Algoritmos** lidam com a **velocidade das operações** e os padrões de projeto lidam com a **velocidade de desenvolvimento**.

PADRÕES DE PROJETO

- **Padrões de criação:** Usado para criar objetos
- **Padrões estruturais:** Padrão associado a estrutura das composições
- **Padrões comportamentais:** Descrevem um padrão de comunicação entre classes e objetos. O ponto principal são os algoritmos e a atribuição de responsabilidades entre objetos.

PROBLEMA?

Padrões de projeto são estratégias genéricas que lidam com problema gerais que surgem na programação orientada a objetos.

Para solucionar esses problemas devemos nos perguntar...

“O que provoca revisão em projetos”?

“O que irá variar em um projeto”?

MATRIZ DE RISCO PARA MONITORAMENTO ESTRATÉGICO DO DISTANCIAMENTO

TAXA DE POSITIVIDADE¹	MUITO CRÍTICA > 70%	RISCO MODERADO	RISCO ALTO	RISCO ALTO	RISCO MUITO ALTO	RISCO EXTREMO	RISCO EXTREMO
	CRÍTICA 61% A 70%	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO EXTREMO	RISCO EXTREMO
	MUITO ALTA 51% A 60%	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	ALTA 41% A 50%	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	MODERADA 31% A 40%	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	BAIXA 21% A 30%	RISCO BAIXO	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	MÍNIMA ≤ 20%	RISCO BAIXO	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	MÍNIMA ≤ 50%	BAIXA 51% A 60%	MODERADA 61% A 70%	ALTA 71% A 80%	MUITO ALTA 81% A 90%	CRÍTICA > 90%	
TAXA DE OCUPAÇÃO GERAL DE UTI ADULTO (PÚBLICO E PRIVADO)							

Percentual de Exames Positivos / Exames Realizados (Síndrome Gripal + Síndrome Respiratória Aguda Grave)

TRÊS TIPOS DE ESTADOS

- Taxas de Positividade
- Taxas de UTI
- Riscos

MATRIZ DE RISCO PARA MONITORAMENTO ESTRATÉGICO DO DISTANCIAMENTO							
TAXA DE POSITIVIDADE¹	MUITO CRÍTICA > 70%	RISCO MODERADO	RISCO ALTO	RISCO ALTO	RISCO MUITO ALTO	RISCO EXTREMO	RISCO EXTREMO
	CRÍTICA 61% A 70%	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO EXTREMO	RISCO EXTREMO
	MUITO ALTA 51% A 60%	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	ALTA 41% A 50%	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	MODERADA 31% A 40%	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	BAIXA 21% A 30%	RISCO BAIXO	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	MÍNIMA ≤ 20%	RISCO BAIXO	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
		MÍNIMA ≤ 50%	BAIXA 51% A 60%	MODERADA 61% A 70%	ALTA 71% A 80%	MUITO ALTA 81% A 90%	CRÍTICA > 90%
TAXA DE OCUPAÇÃO GERAL DE UTI ADULTO (PÚBLICO E PRIVADO)							

Percentual de Exames Positivos / Exames Realizados (Síndrome Gripal + Síndrome Respiratória Aguda Grave)

PADRÕES DE PROJETO

Padrões comportamentais

- **State:** Permitir que um objeto altere seu comportamento quando houver mudança de estado.
- **Strategy:** Criar uma estratégia para diferentes variações que ocorrem em um algoritmo.

Um aspecto importante de qualquer padrão de projeto esta na capacidade de permitir alterações facilmente. Os padrões State e Strategy não são diferente dos demais padrões quando se trata da facilidade das atualizações e mudanças.

PADRÕES DE PROJETO

State – Transição de estados

Um dos problemas com mudanças frequentes de estado em um objeto está na **dependência de instruções condicionais**. Não há nada de errado com as instruções condicionais. Os problemas surgem quando há uma quantidade tão grande de opções a ponto de deixar o programa confuso ou quando elas consomem muito tempo para serem adicionadas ou alteradas e passam a se tornar algo excessivamente oneroso.

PADRÕES DE PROJETO

State – Transição de estados

Todos os padrões State devem ter um participante para armazenar o **estado em que o objeto se encontra**. Essa função pertence a classe contexto que é a classe principal. Teremos uma **classe de contexto para as taxas de positividade e um contexto para as taxas de uti**. Uma classe cliente realizará solicitações por meio da classe de contexto. Não há nenhuma conexão direta entre a classe cliente e qualquer uma das classes de estado.

PADRÕES DE PROJETO

State – Transição de estados

A máquina de estados consiste em um modelo que estabelece o foco em diferentes estados, na transição de um estado para outro e nos acionadores que fazem com que o estado seja modificado. Assim temos **os estados, as transições e os acionadores**.

Com o padrão de projeto State, cada estado possui sua própria classe concreta implementada a partir de uma interface comum. Nesse estudo, os estados são **representados por cada taxa de positividade e por cada taxa de uti**.

PADRÕES DE PROJETO

State – Transição de estados

A classe de contexto cria instancias de implementações das interfaces de estado. O processo de instanciação envolve um tipo de recursão conhecido como autorreferência .

O argumento usado como parâmetro em cada caso é `$this`, que é uma referencia a própria classe de contexto.

Como algum estado deve corresponder ao estado corrente na inicialização, a **propriedade** *taxaAtual* recebe o valor inicial do estado de acordo com o informado pelo usuário. Esse valor corresponde a uma instância de uma das classes de Taxas e o programa segue de acordo esses valores.

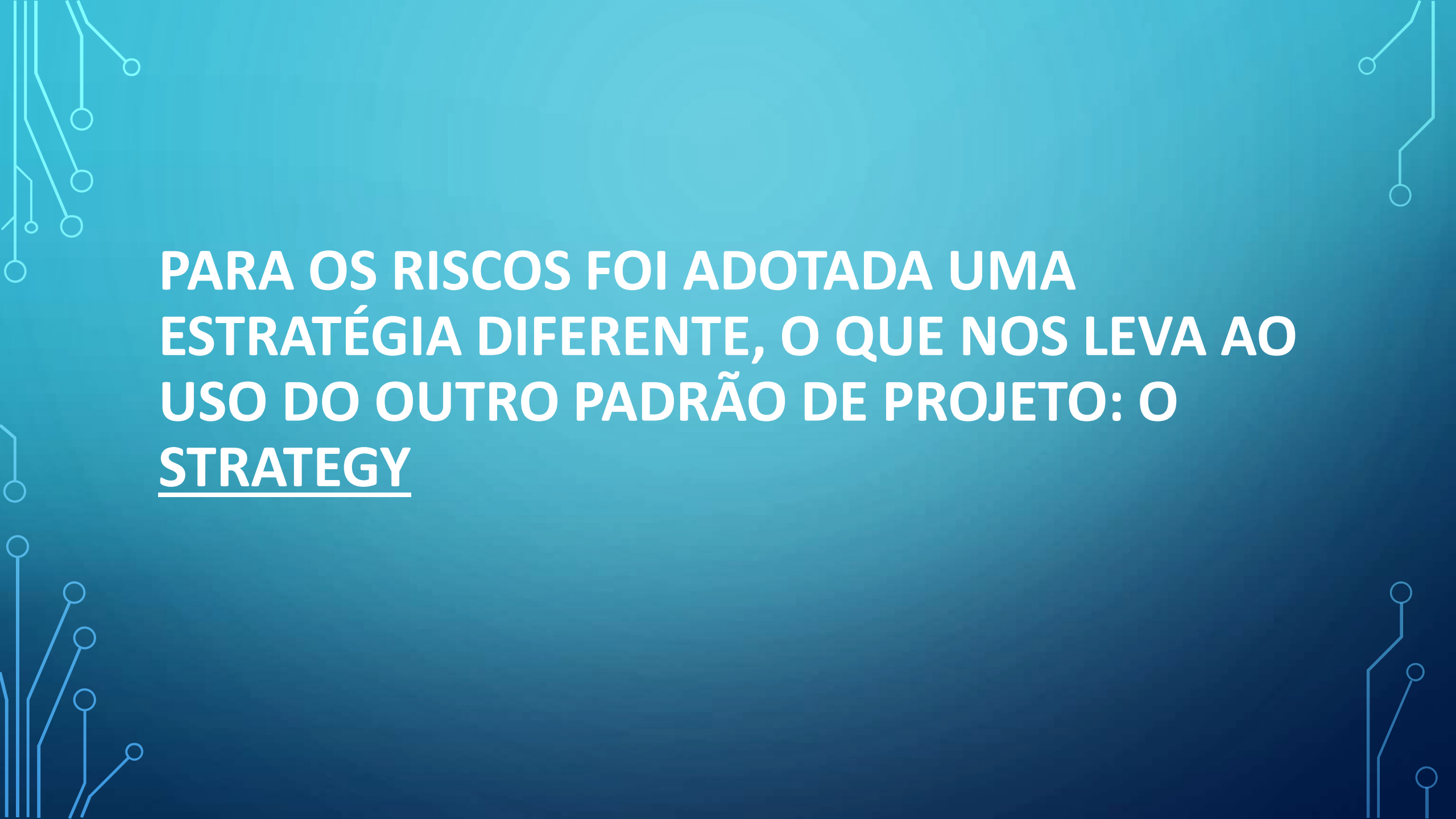
E OS RISCOS TAMBÉM NÃO SÃO ESTADOS?

MATRIZ DE RISCO PARA MONITORAMENTO ESTRATÉGICO DO DISTANCIAMENTO

TAXA DE POSITIVIDADE¹	MUITO CRÍTICA > 70%	RISCO MODERADO	RISCO ALTO	RISCO ALTO	RISCO MUITO ALTO	RISCO EXTREMO	RISCO EXTREMO
	CRÍTICA 61% A 70%	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO EXTREMO	RISCO EXTREMO
	MUITO ALTA 51% A 60%	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	ALTA 41% A 50%	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	MODERADA 31% A 40%	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	BAIXA 21% A 30%	RISCO BAIXO	RISCO MODERADO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
	MÍNIMA ≤ 20%	RISCO BAIXO	RISCO BAIXO	RISCO MODERADO	RISCO ALTO	RISCO MUITO ALTO	RISCO MUITO ALTO
			MÍNIMA ≤ 50%	BAIXA 51% A 60%	MODERADA 61% A 70%	ALTA 71% A 80%	MUITO ALTA 81% A 90%

TAXA DE OCUPAÇÃO GERAL DE UTI ADULTO (PÚBLICO E PRIVADO)

Percentual de Exames Positivos / Exames Realizados (Síndrome Gripal + Síndrome Respiratória Aguda Grave)

The background is a solid blue gradient. In the corners, there are decorative white line art elements resembling circuit boards or neural networks, with lines and small circles connecting them.

**PARA OS RISCOS FOI ADOTADA UMA
ESTRATÉGIA DIFERENTE, O QUE NOS LEVA AO
USO DO OUTRO PADRÃO DE PROJETO: O
STRATEGY**

PADRÕES DE PROJETO

Strategy – Flexibilidade entre as variações

Um dos princípios mais importantes dos padrões de projeto consiste em encapsular o que varia. As variações podem ser pequenas ou grandes, mas ao usar o padrão de projeto Strategy, podemos simplificar enormemente o processo.

PADRÕES DE PROJETO

Strategy – Flexibilidade entre as variações

Com os padrões de projeto a pergunta é? “O que provoca revisão em um projeto”. Assim prosseguimos buscando evitar aquilo que force a necessidade de revisão. Mas e se pensássemos em uma maneira de efetuar alterações sem ter de revisar o projeto? Ao encapsular a parte variável os programadores primeiramente decidem o que irá variar em um programa e então encapsulam esses recursos. Se o projeto exigir mudanças, os elementos encapsulados poderão ser alterados sem que o restante do sistema seja afetado.

PADRÕES DE PROJETO

Strategy – Flexibilidade entre as variações

Mas qual a variação no caso dos riscos? Bom, para saber em qual risco estamos foi definida uma *string* que representa o “eixo” entre o estado resultante da taxa de positividade e o estado resultante da taxa de uti registrada pelo State. Analisando nossa matriz podemos verificar que temos diversos eixos diferentes, as taxas podem ir de taxa de positividade e taxa uti “mínima e mínima” para “muito crítica e crítica”, esses diferentes eixos são as nossas estratégias! Cada risco possui um conjunto de resultados (eixo) que foi construído em um array. Esse array de eixos está representado em cada classe de risco de acordo com as possibilidades na matriz.

PADRÕES DE PROJETO

Strategy – Flexibilidade entre as variações

Com **Strategy** as diferentes tarefas são tratadas por diferentes estratégias concretas e como o cliente solicita estratégias concretas por meio do contexto, ele deve ser conhecedor das estratégias disponíveis. Qualquer projeto baseado em um conjunto de comportamentos pode ser convertido em um padrão de projeto Strategy por meio do encapsulamento dos comportamentos. Os comportamentos exigem algum tipo de algoritmo para torná-los funcionais. Ao encapsulá-los em estratégias concretas eles podem ser usados, reusados e modificados sem causar erros em demais partes do sistema.

PADRÕES DE PROJETO

Diferenças entre State e Strategy – o contexto

A classe de contexto possui uma relação de agregação com uma interface em ambos os padrões.

No padrão State, a relação é com a interface State, e no padrão Strategy, é com a interface Strategy.

PADRÕES DE PROJETO

Diferenças entre State e Strategy

Para compreender a diferença com o State é necessário observar como os diferentes participantes na classe contexto **se comportam no relacionamento com a interface Strategy**.

Os estados concretos disponibilizam métodos para efetuar a transição para outro estado a partir do estado corrente, registrado em uma variável da classe de contexto. Mas o participante “contexto” no padrão Strategy não possui nenhum registro de estratégia corrente em uso. Não há motivos para isso, pois, diferentemente das mudanças de estado, em geral, **a mudança de algoritmos não depende do algoritmo atual em uso**.

PADRÕES DE PROJETO

Classes de contexto e as interfaces

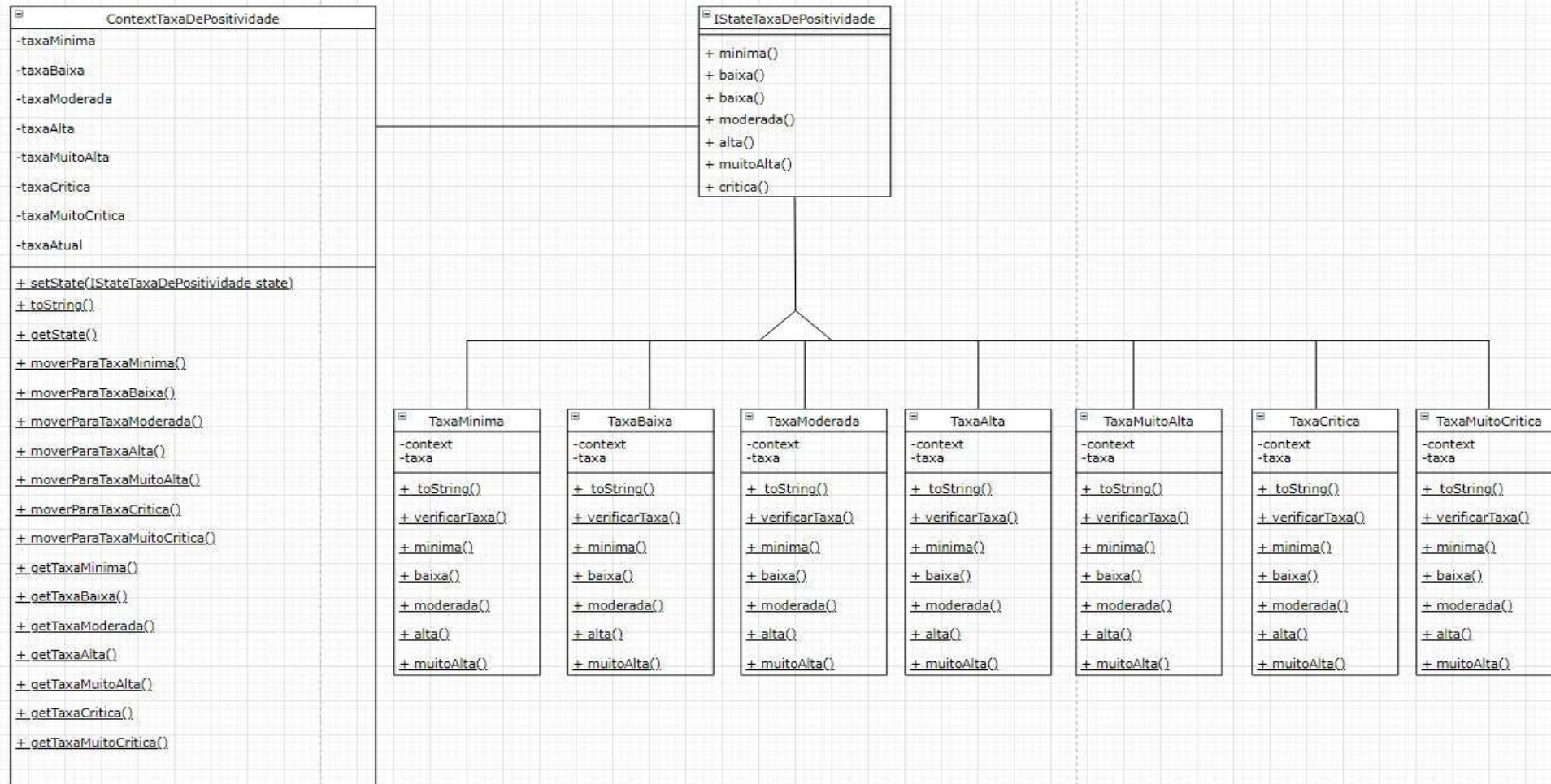
No padrão State, as classes de contexto atua como um armazenados do estado corrente. No padrão Strategy, o contexto tem uma função diferente. Ele serve para separar uma solicitação de uma estratégia concreta, permitindo assim, que a estratégia e a solicitação atuem independentemente uma da outra. Isso representa uma forma de **baixo acoplamento** entre solicitação e consequência.

PADRÕES DE PROJETO

Estrutura e estados para representar a taxa de positividade

- IStateTaxaDePositividade
- ContextTaxaDePositividade
- TaxaBaixa
- TaxaMinima
- TaxaModerada
- TaxaAlta
- TaxaMuitoAlta
- TaxaCritica
- TaxaMuitoCritica

DIAGRAMA TAXAS DE POSITIVIDADE

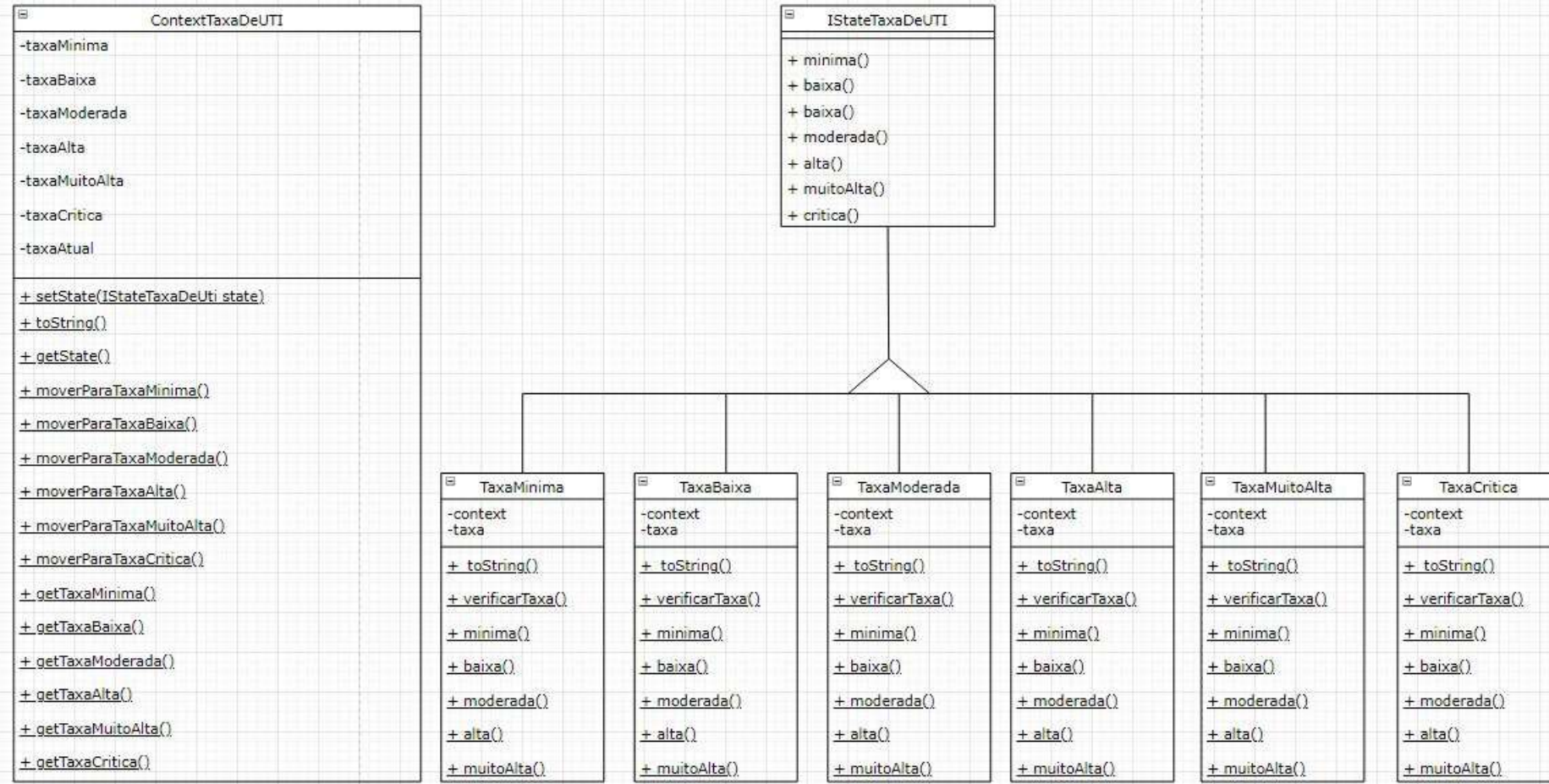


PADRÕES DE PROJETO

Estrutura e estados para representar a taxa de uti

- IStateTaxaDeUTI
- ContextTaxaDeUTI
- TaxaBaixa
- TaxaMinima
- TaxaModerada
- TaxaAlta
- TaxaMuitoAlta
- TaxaCritica

DIAGRAMA TAXAS DE UTI

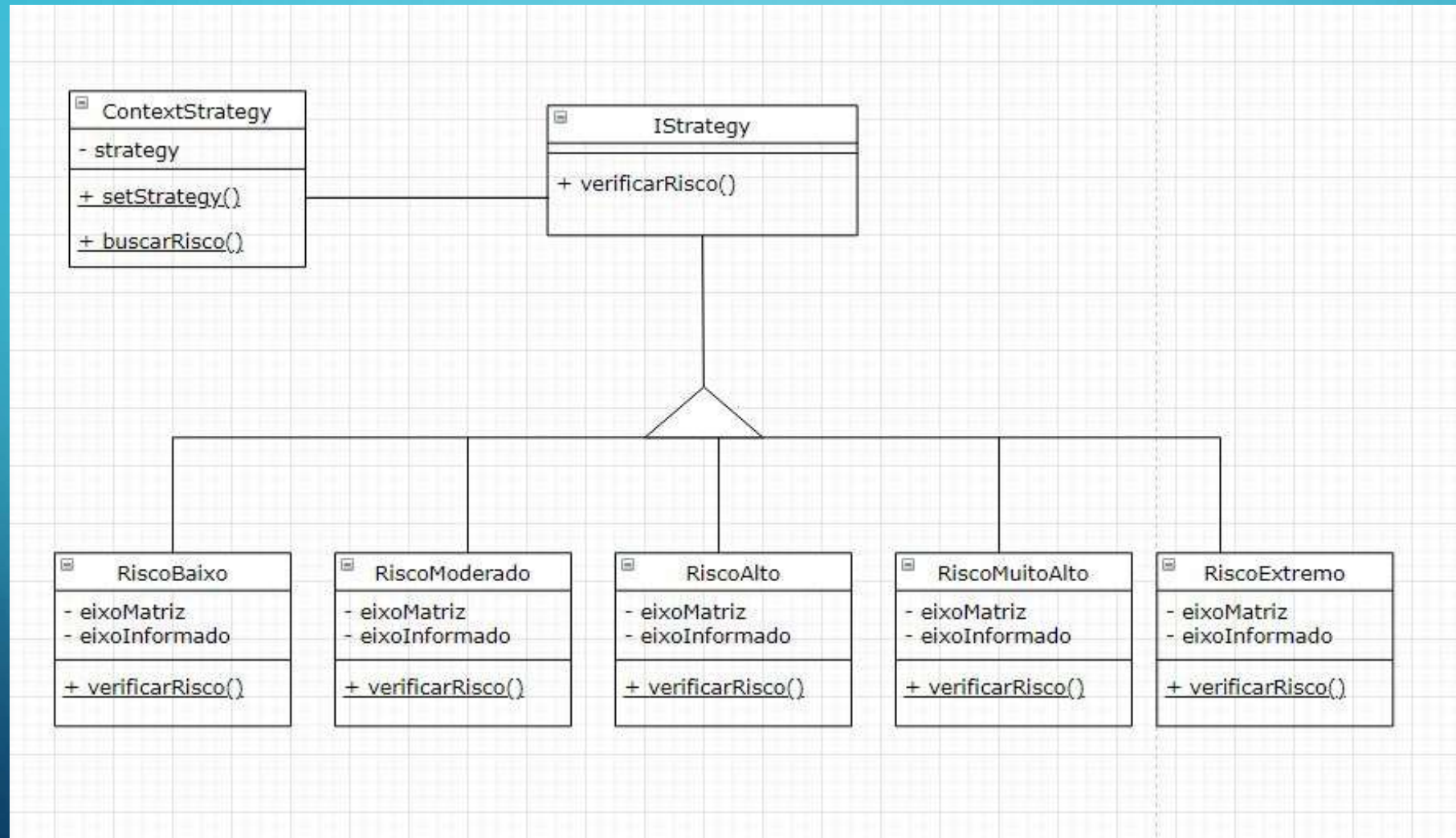


PADRÕES DE PROJETO

Estrutura representar os riscos

- IStrategy
- ContextStrategy
- RiscoBaixo
- RiscoModerado
- RiscoAlto
- RiscoMuitoAlto
- RiscoExtremo

DIAGRAMA DOS RISCOS



PADRÕES DE PROJETO

Não pode usar condicionais em padrões de projeto?

Em nossas construções mesmo assim observamos que existem condicionais. Mas elas não deveriam ser eliminadas?

PADRÕES DE PROJETO

Não pode usar condicionais em padrões de projeto?

Os padrões de projeto não defendem que instruções condicionais jamais sejam usadas, mas em alguns casos elas podem dificultar a manutenção. Se uma única estratégia (algoritmo encapsulado) for alterada e isso exigir a mudança de todo um conjunto de instruções condicionais ou de casos, haverá muito mais chances de erros...

PADRÕES DE PROJETO

Não se pode usar condicionais em padrões de projeto?

... Ao usar qualquer um dos padrões a introdução de condicionais ou de casos no participante do cliente é aceitável porque tudo que o cliente faz é efetuar solicitações. Mesmo nos algoritmos encapsulados a execução de uma tarefa poderá exigir uma instrução condicional.

No **Strategy** as estratégias eliminam as instruções condicionais para a **seleção do comportamento desejado**. No State as instruções condicionais são eliminadas para a **seleção do estado desejado**.

PADRÕES DE PROJETO

Agilidade na administração de grandes sistemas

A partir da problemática na matriz vimos que vários objetos foram criados para lidar com os diferentes estados ou diferentes variações chamados a partir de diversas solicitações de um cliente.

Agora, isso não representa um problema tão sério diante das vantagens relativas a reutilização e a facilidade de mudanças possibilitada por cada padrão.

Os padrões de projeto foram concebidos para agilizar a **administração de uma aplicação** e não para agilizar a execução do código.

AGRADECIMENTOS

PROFESSOR E COORDENADOR DO CURSO EDUARDO BONA

PROFESSOR E ORIENTADOR DESTE PROJETO ER. GALVÃO ABBOTT

A TODA EQUIPE DE COORDENAÇÃO DA PÓS GRADUAÇÃO

FAMILIARES QUE APOIARAM DURANTE O DESENVOLVIMENTO DESTE TRABALHO